PHASE MANIPULATION OF SPEECH USING FIR DIGITAL FILTERS

A thesis presented for the degree of

Doctor of Philosophy

in

Electrical Engineering

in the

University of Canterbury

Christchurch, New Zealand

by

R. D. C. Stephen, B.E. (Hons 1)

1987

ABSTRACT

Three related investigations involving the fields of FIR digital filters, phase manipulation of speech, and speech coding via bandwidth compression are reported.

The first investigation is aimed at providing a means of generating the impulse response coefficients of a non-linear phase FIR digital filter. Existing methods of designing linear-phase filters are discussed and compared from a defined common comparison base.

The methods available for designing non-linear phase filters are examined. An existing linear phase design method is extended to the non-linear phase case and shown to be useful. The required impulse response length in the presence of non-linear phase is studied. Particular emphasis is placed on "random phase" filters and their generation because they are required by the second investigation.

The second investigation examines in detail the ramifications of phase randomising a speech signal. The analytic zero representation of speech which forms the underlying base on which the discussion, and answers, are based is elucidated. The technique of using a non-linear phase FIR filter is shown to be feasible and as a minimum, offers at least the same level of performance as a very early reported technique. Significant differences in the behaviour of male and female speech is demonstrated.

The third and final investigation reports some early and incomplete experiments on a radically different approach to achieving bandwidth compression and expansion of a signal. The technique is referred to as "phase unwrapping". It is based on the application of a linear phase FIR digital filter in an adaptation of the traditional convolution relation. The motivation and validity of the basic idea is outlined and justified via application of the procedure to simple sinusoids and one experiment using real speech. The fundamental problem to be overcome is identified and the basis of a possible means of solution indicated.

ACKNOWLEDGEMENTS

The financial support of the New Zealand Post Office that allowed me to undertake this research is gratefully acknowledged.

I wish to express my gratitude to my supervisor Mr. J.A. Webb. His advice, assistance, encouragement and unfailing optimism are gratefully acknowledged.

In addition, the significant assistance and advice provided by Dr. G.R. Dunlop of the Dept. of Mechanical Engineering and Dr. A.C.J. Hamilton of Post Office Regional Engineers Office, Wellington on some aspects of the work is also gratefully acknowledged.

Discussions with Professor R.H.T Bates, Dr. P. Gough and Mr. W.K. Kennedy of the Department, at various times, are also gratefully acknowledged.

I wish to thank my fellow students for their companionship and, particularly Nigel Brieseman and Richard Lane for their discussions during this research.

Finally, to my wife, Jan, who, in spite of considerable neglect at times, gave continual support and encouragement, thank you.

CONTENTS

CHAPTER 1.0

INTRODUCTION

1.0 PREFACE

The invention of the telephone by Alexander Graham Bell in 1876 gave birth to two diverse, but related, fields of research and endeavour. The first we know today as communications engineering and the second as the analysis, synthesis and perception of the speech signal itself. That a greater understanding of the speech signal and our perception of it should be sought is not surprising for two reasons. First, because if a system is to be near optimal in its operation, it must be designed to match the characteristics of the signals it is to handle and secondly, because such an understanding would allow more efficient use of the physical resources of long distance communication links.

The early electronic communications equipment was entirely analog. So too are the events and phenomena of the world we exist in. The realisation and development of a body of theory that allowed the continuous, analog, events to be described by discrete "snapshots" like the separate frames of a movie film gave rise to a parallel field of study we now describe as digital signal processing. The development of this field was relatively slow until the appearance of the first electronic computers. Since computers work on discrete bits of information, the arrival of computers spurred the development of all aspects of digital signal processing.

Subsequently, it did not take very long to demonstrate that significant advantages could be obtained in maintaining signal quality by representing and transmitting the signals in digital form. Furthermore, using digital signal processing techniques, it was possible to implement certain functions in an accurate, repeatable manner relatively cheaply where before such functions were either only theoretically possible as mathematical representations, or extremely costly and difficult to implement in analog hardware. In recent years, the accumulated body of knowledge and increasing power of computers has made it more economic to simulate and refine the parameters of a system on a computer, before any experimental hardware is built.

The advent of digital communications techniques and the desire to maximise the bandwidth efficiency of such systems by restricting the required bit-rate as far a possible has placed considerable demands on what is known about speech and how it may be manipulated to advantage.

1.1 THE GOALS OF THIS RESEARCH

This thesis is concerned with the digital processing on a computer, in non real-time, of speech by linear filtering. There are two principle goals of this research:

(a) the manipulation of the phase spectrum of speech via the use of non-linear phase FIR digital filters. This is motivated by an experiment on phase manipulation of speech originally conducted in 1961, but which left a number of significant questions unanswered. The purpose here is to identify those questions and examine the ramifications of such manipulation on the characteristcs of real speech.

(b) a description of some early, incomplete, experiments that use a linear phase FIR filter in an adaption of the convolution relation to give bandwidth compression and expansion of a signal. This method is described as "phase unwrapping".

An implied goal of the direction of this research is that the knowledge gained may have some application in reduced bit-rate digital communications systems.

The sections that follow describe the contents and goals of each particular chapter. Since FIR digital filters constitute the means of achieving the intended goals of this thesis, the first part of this work is concerned with aspects related to the design of linear and non-linear phase FIR filters.

1.2 THE CONTENTS OF THE THESIS

1.2.1 Chapter 2: Linear Phase FIR Filters

The field of digital filter design has been the subject of intensive study for decades. Today, a considerable body of knowledge exists on the methods and techniques of designing such filters.

The fundamental problem in digital filter design is satisfying the

Fourier duality constraint between time and frequency. The design process is therefore usually posed as an approximation problem. However, there are many issues involved in the design of digital filters in general. Solving the approximation problem to satisfy a frequency response criterion is only one aspect. Other, equally important issues for example are

(a) computational efficiency: algorithmic complexity, computer time and storage requirements.

(b) finite wordlength effects: introduced by rounding "infinite precison" floating point numbers to fixed digital word lengths. This affects the frequency response and the quantisation noise introduced by the filter.

(c) structural forms: different representations provide alternative ways of looking at the problem and affect (a) and (b).

(d) practical implementation: a different formulation of the problem using a different structural representation can result in reduced hardware complexity.

The work in this chapter is aimed at providing a workable, reliable method of generating the coefficients of the impulse response of a linear phase FIR filter that can be extended to include the non-linear phase case. Thus, the general approach adopted for this chapter is, in the first part, to provide a condensed outline of the subject of digital filtering in general followed by a discussion of the actual performance of several selected algorithms.

The chapter begins by setting out some fundamental definitions of linear systems theory in terms related to filter design. It establishes the two fundamental classes of digital filters: FIR and IIR. The origin of the term "linear phase" and the basic approximation problem to be solved are identified. Then the principle frequency response definitions to be used throughout this and the following chapter, are clarified and related to the common usage in the literature.

Next, the principal design methods for FIR filters are surveyed. No attempt is made to be either complete or exhaustive. Rather, the emphasis is on presenting the broad categories of the design methods and their underlying principles. No account is taken of finite word-length effects or practical implementation aspects because all the work to be reported is performed on a computer. All the

computations are done in single precision floating point representation. There are some special sub-classes of filters which are not considered at all. These are the sampling rate conversion interpolating and decimating filters [Crochiere, 1983b] and half-band and mirror filters.

Comparison of the various methods is treated in section 2.4. However, it is demonstrated that useful comparison based on reported work is difficult. A common comparison base is proposed and used with five selected algorithms as an attempt to bring out the practicalities of the different design methods, rather than a theoretical discussion of the relative efficiencies and properties of the algorithms themselves. The results of the comparisons are given in terms of

- the frequency response achieved,
- the computer time required for a solution,
- computer storage requirements to implement the algorithms.

Finally, section 2.5 revisits three of the algorithms implemented: the McLellan-Parks algorithm, Algazi and Suk's algorithm and Cuthbert's method. In the first, certain practical deficiencies which are not apparent in the reported literature are pointed out. The second method is re-examined from the point of view of simplifying its use and then demonstrating additional behavioural aspects not reported by Algazi and Suk. For Cuthbert's method, the issue of selecting weighting values is examined briefly. The comments are made in the context of their later potential use in the work on non-liner phase filters.

With a few notable exceptions [eg Bozic, 1979; Programs for Digital Signal Processing, IEEE 1979], there are few instances where the literature makes the effort to translate the theory into practice. For a novice, and in the absence of working software, this process of translation constitutes the major work effort.

The outcome of this chapter and chapter 3 is a substantial computer program written in VAX FORTRAN, complete with HELP facility and graphics presentation. The program is capable of designing linear and non-linear phase FIR filters by several different methods. The program will generate low-pass, band-pass and Hilbert versions of these filters with odd or even sequence lengths for orders upto 512 in some cases and greater in others when the FFT length used becomes the limiting factor. Special features such as compensating for the sinc function roll-off of a sample-and-hold digital-to-analog converter

have also been incorporated.

1.2.2 Chapter 3: Non-Linear Phase FIR Filters

This chapter expands on the work of chapter 2 to consider the non-linear phase case. In contrast to the case for linear phase filters, very little literature exists on methods for designing non-linear phase filters. Very likely this is due to the fact that such a requirement is completely contrary to the usual practice that requires the linear system (the filter) introduce the least amount of inband distortion (magnitude AND phase) of the signal. A major exception to this rule is the requirement for equalisers in communications systems. Equalisers are a special case of non-linear phase filters called all-pass networks. Such networks have a magnitude response of unity and minimum delay - ie minimum phase (shift). Traditionally, such networks are designed as recursive structures [Brophy et al,1975; Gregorian et al, 1978].

The chapter opens by demonstrating how the basic approximation problem is complicated by the introduction of a non-linear phase response.

Next, existing approaches to the design problem for FIR non-linear phase filters are described. These amount to using three different algorithms: a numerical optimisation algorithm; the Remez exchange algorithm and a non-linear programming problem converted to a linear problem and solved using the simplex optimisation algorithm.

Following this, the discussion digresses to consider the issue of the impulse response length required to satisfactorily approximate the desired frequency response in the presence of a non-linear phase. There are two reasons for this; the first is that it is not necessarily obvious that, in general, a non-linear phase response will require longer impulse response lengths than the equivalent linear phase filters to match the same magnitude response requirements. The second reason is because the information gained in examining this issue for two types of phase response will be used later in the discussion on random phase filters. The two phase responses used for the discussion are a sine deviation and a square-wave deviation. The effect of windowing in the presence of a non-linear phase is also examined and it is demonstrated that the window technique of Chapter 2 is not feasible in the non-linear phase case.

Using the square-wave phase deviation for discussion, the next section reports on the implementation of two of the known design methods. One attempt tries to use existing published software and fails for reasons not properly established. The remaining method uses Cuthbert's method employing Fletcher's numerical optimisation algorithm.

Subsequently, the extension of Algazi-Suk's iterative algorithm of Chapter 2 to the non-linear phase case is considered and presents some examples of its performance.

In Chapter 5, random phase filters will be required. Section 3.6 examines the issues involved in generating the random phase filters. The most important criteria for these filters is that they have small passband ripples to aviod aplitude distortion of the speech signal. This work will build on the earlier discussion of the square-wave phase deviation. It will be shown that the desired form of phase response is practically almost impossible to satisfy and the desired phase response will have to be sacrificed to allow a physically practical filter to be designed and to minimise the passband ripple.

1.2.3 Chapter 4: The Speech Signal

The speech signal - its production, properties and perception - have been studied for decades and a large body of knowledge exits. Flanagan's text [1972] is one good example of the accumulated knowledge, while that of Rabiner [Rabiner et al, 1978] emphasises the techniques of digital processing of the speech signals.

The intention of this chapter is to elucidate the existing essential knowledge of this signal to establish an adequate framework for the work of chapters 5 and 6. The discussion in sections 4.1 and 4.2 is very much a precis and draws heavily on the two texts mentioned above. Rather than a long, detailed description, the approach taken is a descriptive one, with the emphasis on establishing the basic characteristics.

Section 4.6 introduces the mathematics of the Analytic vector and zero distribution representations of a speech signal. These representations have been known and used by others in the past. They are introduced here because they form the basis for the discussion of Chapter 5. Two examples are given to clarify the terms used.

1.2.4 Chapter 5: Speech Phase Randomisation

The work of this chapter is concerned with the phase manipulation of, in particular, voiced speech. The stimulus for the work reported here is an early experiment on the phase manipulation of real speech by David et al [1961]. Rather than attempt to corroborate or argue the case about phase perception of the ear, the emphasis here is an attempt to answer some implied questions raised by this early experiment using the work of Hamilton [1985] as a base and, if possible, taking advantage of digital signal processing techniques that were not possible at the time of the original experiment, either through technological limitations, or the lack of knowledge.

The chapter opens by describing the original experiment and subsequently identifies important questions that arise. Preparatory to finding answers, section 5.3 describes the generation of the speech data base on the computer, the principle of the simulations and the main objective measure called Peak Factor that will be used to illustrate the effects of manipulating the phases of the signal.

The approach to suggesting some answers uses the mathematical representations of the voiced speech signal outlined in Chapter 4. The idea of using the zeros of a signal in general [Bond et al, 1958; Voelker, 1966a] and speech in particular [Morris, 1972] as a "shorthand" description of the signal is not new. The principal problem for those unfamiliar with this form of description is in establishing a connection between the zeros and the time behaviour of the signal. However, it is this relationship that provides the key to some of the questions raised. Thus, the discussion of the amplitude and zero structure of voiced speech begins by analysing a simple synthetic signal of 3 harmonics as the means to justifying the underlying characteristics of the voiced segments of speech.

The requirements of the non-linear phase characteristics of the filters are outlined next, based on the established understanding of the principle characteristics of voiced speech established in chapter 4. In doing so, it explains the emphasis of the discussion on the design aspects of such filters in Chapter 3.

The analysis of aspects relating phase randomisation of real speech begins with an analysis of a single word by building on the initial 3-sinusoid example. The discussion using the 3-sinusoid model

raises an important question suggesting that a signal with the minimum peak factor contains only real zeros. This aspect is considered briefly where the discussion is deliberately careful to avoid proposing "real" answers. The sole reason for this care is because the implications of this requirement of the signal zeros occurred late in the work and insufficient rigorous analysis has been performed.

Following this, real conversational speech of approximately 5 seconds duration from two male and two female adult speakers is processed and examined from four points point of view: change in

- peak factor,
- signal dynamic range,
- perceptual alteration in quality and intelligibility and
- amplitude probability density function

Finally, some answers to the questions originally posed are suggested.

1.2.5 Chapter 6: Phase Unwrapping of Speech

This chapter introduces some initial experiments on attempts to halve the bandwidth of a signal by applying the coefficients of an ordinary linear phase FIR filter in an adaption of the traditional convolution process. The intention of this method is that it can be used to provide bandwidth compression of a speech signal at a lower level of complexity than existing methods.

The experiments are incomplete simply because the idea for this approach occurred late in the work and the writer simply ran out of time to pursue the subject in more detail. Thus, no attempt is made to be complete. The emphasis of the work reported here is to describe what has been achieved so far, and to relate it to the known methods of frequency division.

The chapter opens with a very brief introduction to the broad subject of digital speech coding. The intention is simply to put the following sections in context, rather than a detailed summary of the multitude of different algorithms and philosophies. This is followed by a section on known methods of achieving frequency division and serves the same purpose, but is more important here because it allows the phase unwrapping method to be categorised in relation to the known methods.

The principle of applying a FIR filter to divide and subsequently expand a signal is then described. First, the basic idea which gave rise to the experiments is outlined. Then the actual experiment that give a division (and subsequenct expansion) by 2 is described. For clarity and to be able to clearly establish what is happening, single or two-tone sinusiods are used intially as the input signals.

Finally, the fundamental problem that must be overcome is identified and one approach that has been considered as a means of overcoming the problem is outlined.

1.2.6 Chapter 7: Conclusions and further research

This chapter concludes the thesis by highlighting the principal outcome of this work and in so doing identifies additional questions that remain unanswered.

1.3 PRELIMINARIES

Since this work utilises digital signal processing techniques with sampled data, the brief sections that follow establish the principle aspects of Fourier theory that form the underlying base of this entire work. As such, it provides a convenient opening for Chapter 2.

1.3.1 The Fourier Transform and Sampling

The Fourier Transform [Papoulis, 1962a] $F(\omega)$ of a continuous, real function $f(t)$ is defined as

$$F(\omega) = \int_{-\infty}^{\infty} f(t)e^{-j\omega t}\, dt \qquad (1.1)$$

and its inverse as

$$f(t) = \frac{1}{2\pi}\int_{-\infty}^{\infty} F(\omega)e^{j\omega t}\, d\omega \qquad (1.2)$$

and the notation $f(t) \longleftrightarrow F(\omega)$ denotes a Fourier Transform pair. $F(\omega)$ is in general complex:

$$F(\omega) = F_r(\omega) + jF_q(\omega) \qquad (1.3a)$$

$$= A(\omega)e^{j\phi(\omega)} \qquad (1.3b)$$

$A(\omega)$ is the magnitude, $\phi(\omega)$ the phase and ω is the continuous (radian) frequency: $\omega = 2\pi f$. For $f(t)$ a real function, the components of $F(\omega)$ exhibit Hermitian symmetry:

$$F_r(\omega) = F_r(-\omega) \quad \text{, even function} \tag{1.4a}$$

$$F_q(\omega) = -F_q(\omega) \quad \text{, odd function} \tag{1.4b}$$

Note in the above discussion, that the Fourier Transform and its inverse requires the defining function be known over the infinite range $(-\infty,\infty)$ and that apart from requiring that $f(t)$ be real and square integrable, no special properties are ascribed to it.

For a discrete signal, the development of its Discrete Fourier transform (DFT) has some important consequences on the properties of the functions that form a Discrete Fourier transform pair. Fig. 1.1 gives a pictorial development of the DFT [Brigham, 1974]. Denoting * as the convolution operator, the well known convolution theorem can be expressed in two ways:

time convolution: $$f_1(t)f_2(t) \longleftrightarrow F_1(\omega)*F_2(\omega) \tag{1.5a}$$

frequency convolution: $$f_1(t)*f_2(t) \longleftrightarrow \frac{1}{2\pi} F_1(t)F_2(t) \tag{1.5b}$$

Defining the impulse function

$$\delta(t) = \int \delta(t-t_0)f(t) = f(t_0) \tag{1.6}$$

then the sampling function $s_T(t)$ and its Fourier transform are (Fig. 1.1(b) and 1.1(c))

$$s_T(nT) = \sum_{n=-\infty}^{\infty} \delta(t-nT) \longleftrightarrow \frac{2\pi}{T} \sum_{n=-\infty}^{\infty} \delta\left(\omega - \frac{2\pi n}{T}\right) \tag{1.7}$$

where T is the sampling interval, so that, using equation (1.5a), the sampled version of $f(t)$ is $f(t)s_T(t)$ which has the spectrum of Fig. 1.1c. Observe that the spectrum is periodic and the overlap, or "aliasing", of the spectrum at the 1/2T points is a manifestation of the Sampling Theorem that requires the minimum sampling rate of an analog signal occur at a rate at least twice the highest frequency component of $f(t)$. ie

$$\omega_s \geq 2\omega_m \tag{1.8}$$

Thus, the first consequence of discretising $f(t)$ is that $f(t)$ must, in

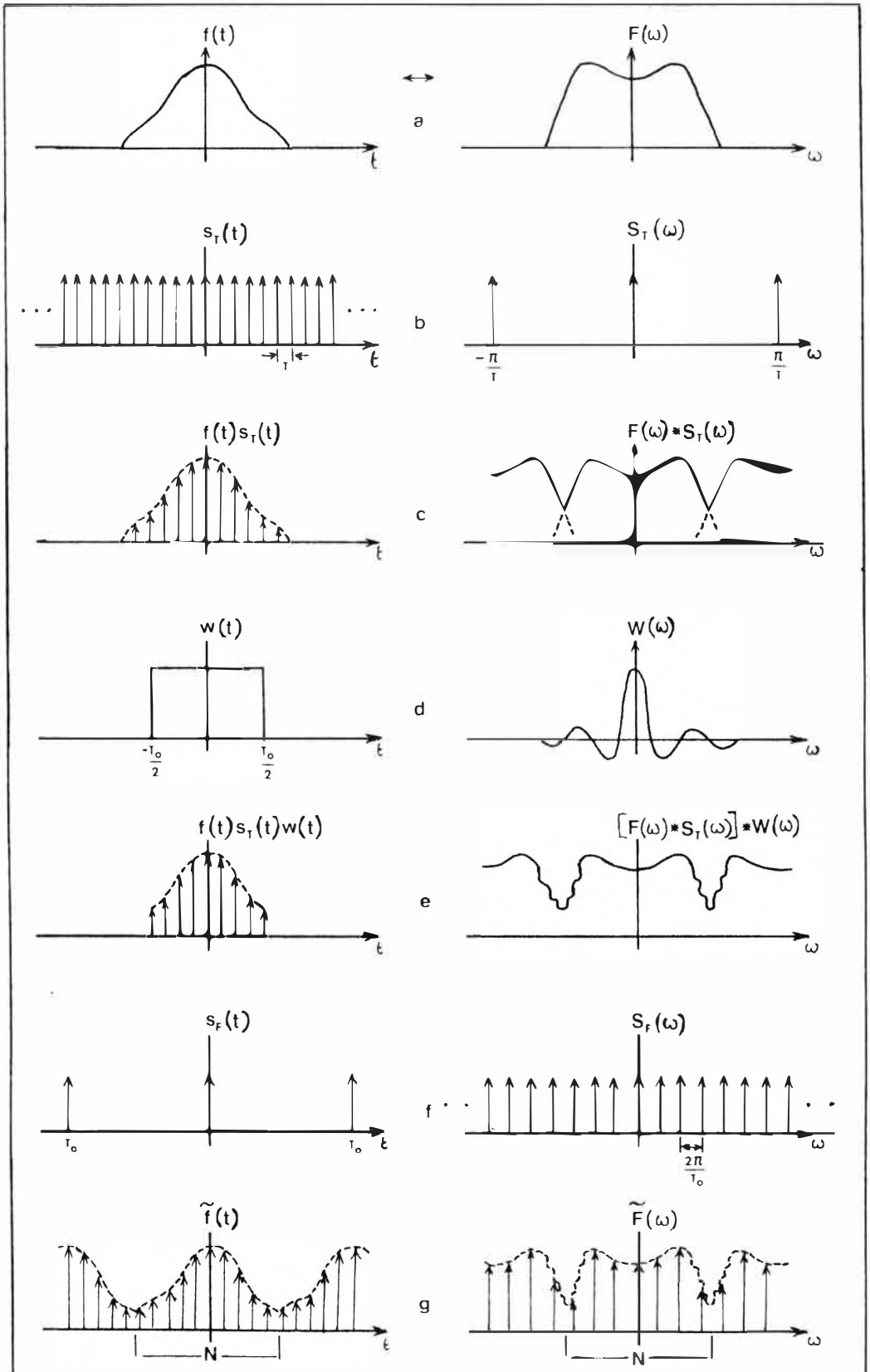


Fig. 1.1: The development of the Discrete Fourier Transform from its continuous definition at a.
[after Brigham, 1974]

practice, be strictly bandlimited if the aliasing of the spectrum is to be avoided. Also note that at this point, the sampled function f(t) is defined for all time and has a continuous spectrum.

Practically, f(t) is only ever known over a finite interval. The penalty incurred by this restriction is the introduction of ringing or ripple in the spectrum of f(t). Truncating (or "windowing") the infinite (time) length by the "window" of Fig 1.1(d) restricts the extent of f(t) to a (time) length T_0 - Fig 1.1e.

The final step is to discretise the frequency spectrum by introducing the frequency sampling function $S_F(\omega_k)$

$$S_F(\omega) = \frac{1}{2\pi} \sum_{k=-\infty}^{\infty} \delta(\omega - k\omega_0) \ , \quad \omega_0 = \frac{2\pi}{T_0} \tag{1.9}$$

which, from equation (1.7) and shown in Fig 1.1(f) gives a time spacing of the impulses of T_0. The consequence of the frequency sampling function is that the sampled version of f(t) is now periodic in T_0. The final discrete functions are shown in Fig. 1.1(g). Comparing Fig 1.1(b) with Fig. 1.1(f), it is apparent that $N = T_0/T$.

From the development of Fig 1.1, the Discrete Fourier transform and its inverse are

$$F(k) = \sum_{n=0}^{N-1} f(n) e^{-j\frac{2\pi kn}{N}} \tag{1.10a}$$

$$f(n) = \frac{1}{N} \sum_{k=0}^{N-1} F(k) e^{j\frac{2\pi kn}{N}} \tag{1.10b}$$

where n is the time index and k the frequency index corresponding to dt and dω respectively in equation (1.1) and (1.2). Equation (1.10) is the Fourier transform pair used in practice (on a computer) to handle samples of real data. The time required to compute either of the above equations is proportional N^2. The Fast Fourier Transform (FFT), first published by Cooley et al [1965] is a fast means of computing equation (1.10) where the time required is proportional to $N\log_2 N$.

Note that since the DFT requires both f(n) and F(k) be periodic, using the theory of Fourier series, f(n) can be represented as a Fourier series over the fundamental period:

$$f(n) = \sum_{k=0}^{N-1} A_k e^{j\omega kt} \quad , t = nT, \quad n = 0,\ldots,N-1 \tag{1.11}$$

Equation (1.11) is often used in the literature on FIR filters, where f(n) is called the impulse response and the A_k are the Fourier coefficients describing the frequency response of the filter at the (discrete) frequencies k. This is a Fourier series representation and should not be confused with the inverse DFT of equation (1.10b).

1.3.2 Practicalities in using the DFT

For notational convenience and retaining the link to the original definition of the Fourier Transform, the development of the DFT in Fig 1.1 deliberately centered the functions about the origin. In practice, the principle of causality, $f(t) = 0$, $t<0$ means that information is only available for $t \geq 0$. This is why equation (1.10) above is defined for $n,k = 0,\ldots,N-1$. Practically, this means that the data segment computed is shown in Fig. 1.2. Given the periodicity of the functions, this means that the data in the upper half is equivalent to the negative data in the principal interval of Fig. 1.1(g) and the midpoint (N-1)/2 is the aliasing or "reference" point in the data. This apparent shift of the reference point is important when trying to establish the phase function $\phi(\omega)$ of equation (1.3b) because of the difference between an arbitrarily defined "zero" point of the data and the zero point of the DFT. The work in Chapter 5.0 explicitly compensates for this difference.

The final point related to the use of the DFT is based on the fact that most FFT routines use a highly composite number that is usually

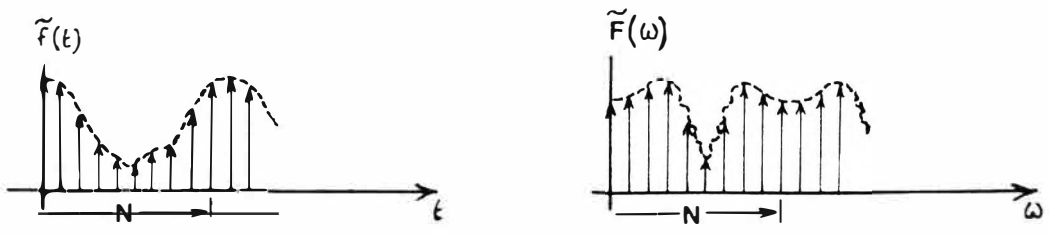


Fig. 1.2: The actual data segment represented and manipulated in practice by the DFT on a computer.

based on a power of 2: $N=2^p$, p an integer. This means that N is an even number. In the frequency domain (index k), k=0 corresponds to the DC term. This leaves an odd number to represent the spectral information over one period and establish the aliasing point. The approach adopted throughout this work is to assign, in the frequency domain:

$k = 0$: DC term

$k = 1,..,\frac{N-1}{2}+1$: "+ve harmonics"

$k = \frac{N-1}{2}+2,...,N-1$: "-ve harmonics"

This assignment always provides one more "+ve" harmonic than there are "-ve" harmonics.

CHAPTER 2.0

LINEAR PHASE FIR FILTERS

2.1 INTRODUCTION

This chapter presents a survey of the broad classes of design methods of FIR filters and selects five algorithms to implement and compare. A great many different algorithmic approaches have been proposed in the literature. Some are very elegant and mathematically efficient, but require considerable mathematical manipulation to get the algorithms to the point where they can be used on a computer. Thus the discussion that follows is restricted to those that a novice, in the absence of working software, can successfully implement in a reasonable time frame. The emphasis is on trying to put the different algorithmic approaches in perspective from a practical - or users - point of view.

2.2 FUNDAMENTALS

2.2.1 Linear systems Theory

A digital filter is a member of the general class of linear discrete systems. To be realisable, the linear system is causal - ie the output y(n) of discrete system only exists for $n \geq 0$. The general mathematical theory of linear discrete systems is well documented [eg Rabiner et al, 1975a; Oppenheim, 1975]. Suffice to say here that a general expression for an N-th order causal, time invariant, linear discrete system is

$$y(n) = \sum_{i=0}^{N} b_i x(n-i) - \sum_{i=1}^{M} a_i y(n-i) \quad , \quad n \geq 0 \qquad (2.1)$$

where x(n) is the input sequence and y(n) the output sequence. Using the z-transform, such a system can also be described by a rational polynomial in z^{-1} with the transfer function:

$$H(z) = \frac{Y(z)}{X(z)} = \frac{\sum_{i=0}^{N} b_i z^{-i}}{\sum_{i=0}^{M} a_i z^{-i}} \quad , \quad a_0 = 1 \qquad (2.2)$$

Equation (2.2) indicates the system function is characterised by its set of zeros (the numerator) and its set of poles (the denominator). Filters that contain both poles and zeroes - ie they rely on past values of inputs and outputs to compute the present output are recursive and are described in the literature as Infinite Impulse Response (IIR) filters. Such filters possess "infinite memory" in the same manner as an analog R-C filter.

Filters that contain only zeroes in their transfer function - ie rely on the past and present inputs to compute the present output are non-recursive and are described as Finite Impulse Response (FIR) filters. This principle distinction has resulted in two entirely separate classes of filters, each with its own attributes and body of literature.

For the purposes of this work, the major differences between the two types are that FIR filters are guaranteed to be always stable and can have exact linear phase. IIR filters can suffer instability and limit cycles, especially when the coefficients of the impulse response are quantised, and exact linear phase cannot be guaranteed. This work is about FIR filters, so all further mention of IIR filters will be dropped from this point.

For a FIR system, equation 2.1 reduces to

$$y(n) = \sum_{i=0}^{N} b_i x(n-i) \qquad (2.3)$$

This relationship between input and output can be represented by many structures [Oppenheim, 1975b]. Different structures may be equivalent from some points of view, but not others - eg when finite word-length effects are considered. The direct form structure of equation (2.3) is shown in Fig. 2.1 and is the structural form that will be used throughout this work.

The b(i) are the weighting or impulse response coefficients of the system, z^{-1} is a unit delay operator and N is the order of the filter. The work that follows is directed at the methods used to generate the b(i), which, in the context of filter design literature, uses the symbol "h" in place of "b" .

Thus, if h(n) is a real, causal sequence representing the impulse response of a real FIR filter, the frequency response of the filter is

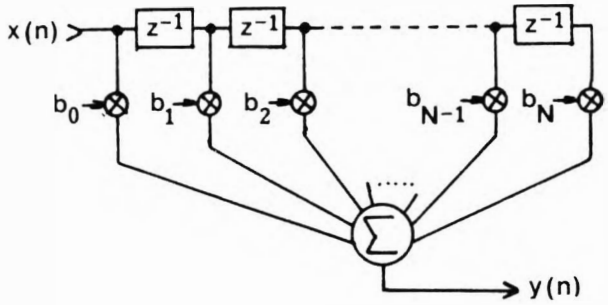


Fig. 2.1: Direct Form Structure that realises the non-recursive difference equation(2.3).

given by the Discrete Fourier transform $H(\omega)$ of $h(n)$:

$$H(\omega) = \sum_{n=0}^{N-1} h(n)e^{-j\omega Tn} \quad , \; n=0,\ldots,N-1 \tag{2.4}$$

where T is the sampling interval of the sequence h(n). At this point, no special properties are ascribed to h(n). Thus, $H(\omega)$ is, in general, complex

$$H(\omega) = H_r(\omega) + jH_q(\omega) \tag{2.5a}$$

$$= |H(\omega)|e^{j\theta(\omega)} \tag{2.5b}$$

Fundamental Fourier theory requires $N \to \infty$ in (2.4) if h(n) is to represent exactly some ideal frequency response - eg the lowpass response of Fig. 2.2.

The description of the frequency response in equation (2.4) reflects the causality requirement of the sequence h(n). By straightforward application of the time-shift property of the Fourier transform by writing k=n-(N-1)/2 in equation 2.4, whence

$$H(\omega) = e^{-j\omega(N-1)T/2} \sum_{k=-\frac{(N-1)}{2}}^{\frac{N-1}{2}} h\left(k+(N-1)/2\right)e^{-j\omega Tk} \tag{2.6}$$

where the "linear phase" term $\phi(\omega)=-\omega T(N-1)/2$ is now clearly evident, and H(w) has real and quadrature components as in (2.5). Using equation (2.5b) in (2.6) and re-arranging,

$$|H(\omega)|e^{j\theta(\omega)}e^{j\omega(N-1)T/2} = \sum_{k=-(N-1)/2}^{(N-1)/2} h\left((k+(N-1)/2\right)e^{-j\omega Tk} \tag{2.7}$$

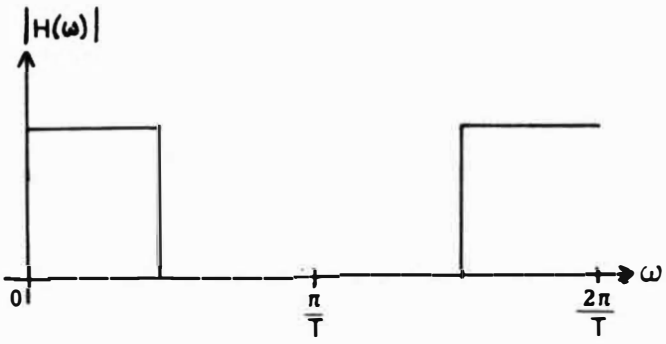


Fig. 2.2: An ideal sampled low-pass filter frequency response. T=sampling interval.

The total phase response of the filter is

$$\Phi(\omega) = \omega(N-1)T/2 + \theta(\omega) \qquad (2.8)$$

If the constraint $H_q(\omega)=0$ in equation (2.5) is imposed, then $\theta(\omega)=0$ and the filter itself has zero phase shift from the linear phase base. Recalling the definition of Group Delay is the delay imparted to the energy (envelope) of a signal:

$$GD = -d\phi(\omega)/d\omega$$
$$= -(N-1)T/2 \qquad (2.9)$$

if $\theta(\omega)=0$

Equation (2.9) show that all frequencies have identical time delay imparted to them so there is no phase distortion of the signal. Thus, the term "linear-phase filter" is used to describe the specific case $H_q(\omega)=0$ and is an implicit statement that the impulse response of the filter is causal - ie it is physically realisable.

2.2.2 Problem Definition

Since N above is finite and usually small, the design problem is an approximation problem: find a design method that will produce a sequence h(n) possessing a frequency response that is the best (in some sense) approximation of the "ideal" or "desired" frequency response. What is "best" depends entirely on the problem at hand and the point of view of the designer. More generally, the finite length of N - ie the time length of the filter - introduces a fundamental limitation which cannot be avoided. This is the time-bandwidth

uncertainty relation which says [Woodward, 1953, p 119; Temes et al, 1973] that if a signal f(t) (or system function) exists over a finite time interval t, and contains a frequency spread $\Delta\omega$, then the condition

$$\Delta\omega\Delta t \geq 1/2 \qquad (2.10)$$

holds. Slepian [1976] descibes this argument rigorously. The design of digital filters is thus a sub-set of the larger general problem of the optimisation of band-limited signals and systems. This larger problem is thoroughly discussed by Papoulis [1967] and Temes et al [1973].

All the design methods that will be discussed are in general manipulating the fundamental limitation of (2.10) to best satisfy one or more particular requirements of the designer, where "best" is most usually understood to be the conflicting requirements of smallest transition width, least passband ripple and stopband magnitude for a chosen value of N that is as small as possible.

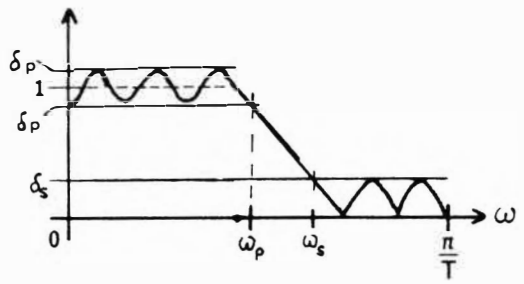
2.2.3 Frequency Response Definitions

A common practice in the literature for defining key parameters in the frequency response of a digital filter is shown in Fig. 2.3(a) for a lowpass filter.

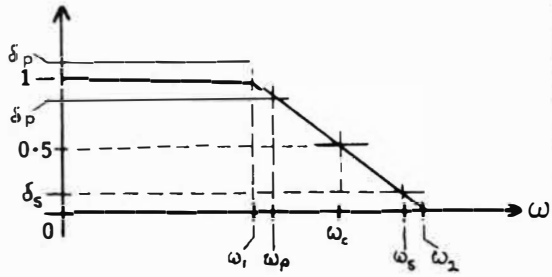
ω_p and ω_s are passband and stopband edge frequencies; δ_p and δ_s are the peak passband and stopband ripple magnitudes. The interval $\omega_s - \omega_p$ is the transition bandwidth where the response is not usually specified, or left as a free parameter in the design algorithm. It reflects the admission above that for a finite N, the ideal response of Fig. 2.2 where $\omega_p = \omega_s$ is not possible.

Such a procedure is useful for mathematical reasons related to optimisation techniques, but conflicts with the well established practice that the cut-off frequency of an (analog) filter is the -3dB point in the magnitude of the frequency response.

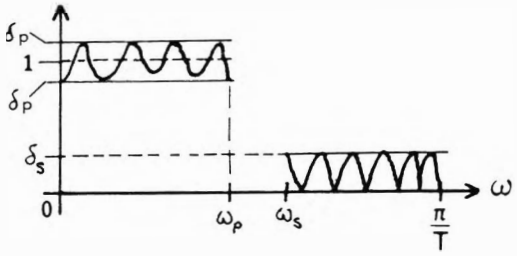
For reasons that will become apparent when the results of some design methods are compared and for consistency among the design methods, all the filters used in this work will be specified using ω_c according to Fig. 2.3(b). The transition band $\omega_1 - \omega_2$, when specified, will be centered symmetrically on ω_c. For bandpass filters, there is one additional specification: ω_0, the centre frequency of the passband.



(a) usually found in the literature



(b) common definition adopted in this work.



(c) McLellan-Parks algorithm only.

Fig. 2.3: Different definitions of frequency response parameters.

The single exception to this rule will be the Mclellan-Parks algorithm. This algorithm requires the specification on a disjoint frequency interval shown in Fig 2.3(c). For this algorithm only, $\omega_1=\omega_p$, $\omega_2=\omega_s$ in the specification of the "desired" frequency response.

2.3 LINEAR PHASE DESIGN METHODS

Fundamental Fourier theory requires that if h(n) in equation (2.4) is real and causal, and $H_q(\omega) = 0$ in equation (2.5), then $H_r(\omega)$ and h(n) possess even symmetry:

$$H_r(\omega) = H_r(\frac{2\pi}{T} - \omega) \quad , \omega = 0,\ldots,\pi/T$$

$$h(n) = h(N-1-n) \quad , n = 0,\ldots,(N-1)/2 \qquad (2.11)$$

However, using the Hilbert Transform defined as

$$\hat{f}(t) = \frac{1}{\pi} \int_{-\infty}^{\infty} \frac{f(t)}{t-\tau} \, d\tau \tag{2.12}$$

It can be shown that

$$\hat{H}(\omega) = -j\,\mathrm{sgn}(\omega)H(\omega) \tag{2.13}$$

$$\mathrm{sgn}(\omega) = \begin{cases} 1 \,, & \omega > 0 \\ 0 \,, & \omega = 0 \\ -1, & \omega < 0 \end{cases} \tag{2.14}$$

which requires in equation (2.5a) that

$$H_r(\omega) = 0$$

$$H_q(\omega) = -H_q\left(\frac{2\pi}{T} - \omega\right) \,, \quad \omega = 0, \ldots \pi/T \tag{2.15}$$

and h(n) is still real and causal, but now possesses odd symmetry:

$$h(n) = -h(N-1-n), \quad n = 0, \ldots, (N-1)/2 \tag{2.16}$$

Fig. 2.4 shows the (non-causal) frequency response of equation (2.13). Thus, there are four particular cases to be considered:

N EVEN

Case 1: $h(n) = h(N-1-n)$, even symmetry (2.17a)

Case 2: $h(n) = -h(N-1-n)$, odd symmetry $n=0,\ldots,(N/2)-1$ (2.17b)

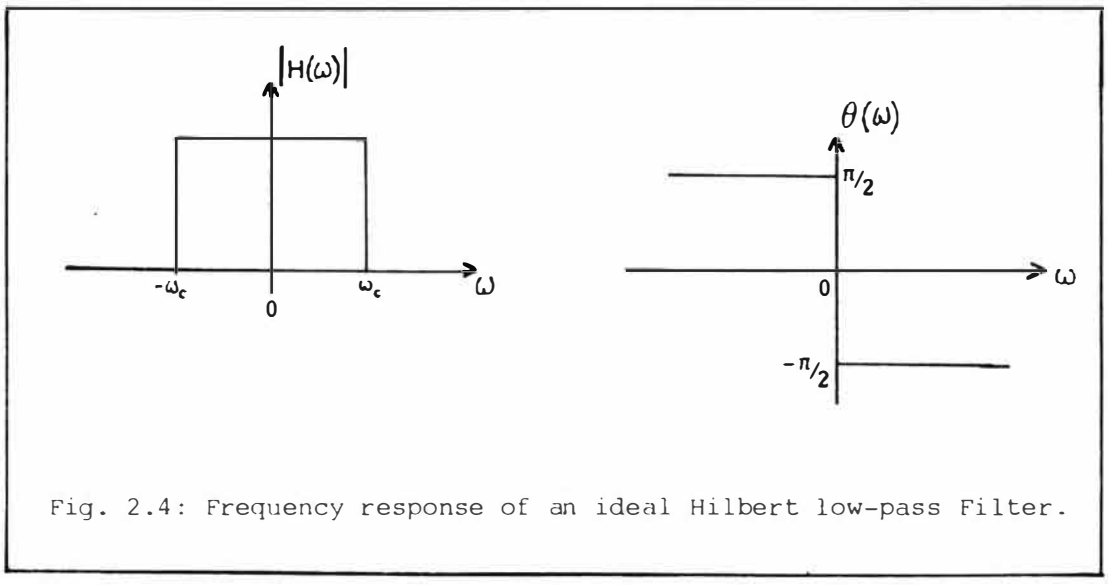


Fig. 2.4: Frequency response of an ideal Hilbert low-pass Filter.

N ODD

Case 3: $h(n) = h(N-1-n)$,even symmetry, $n=0,\ldots(N-1)/2$ (2.17c)

Case 4: $h(n) = -h(N-1-n)$,odd symmetry, $n=0,\ldots(N-3)/2$
$= 0$,$n=[N-1]/2$ (2.17d)

Cases 1 and 3 correspond to basic linear phase filters. Cases 2 and 4 correspond to Hilbert (linear-phase) filters.

It is therefore immediately apparent that whatever the design method chosen, the computational load is reduced by a half because at most only half of the sequence needs to be computed. Application of the symmetry property completes the required sequence.

A broad categorisation of design methods which represent the initial approach to the problem is

(a) zero location and z-transform methods.

(b) energy concentration and smoothing windows.

(c) optimisation (in a mathematical sense) methods.

2.3.1 Zero methods

These techniques are particularly abstruse because they rely on the designer having an extensive experience in the use of z-transform calculus to relate zero positions to frequency responses. This approach has been considered by Requicha and Voelker [1970], while the work of Rader and Gold [1967] clearly demonstrates the fundamental role played by the z-transform in filter design. Bearing in mind that the ultimate interest is in non-linear phase filters (with a particularly difficult phase response), it is considered to be a significant problem in itself to establish the required zero positions of the non-linear phase response in conjuction with the magnitude response. Thus, this approach will not be considered any further.

2.3.2 Energy Concentration - windows

This approach is based on the application of a non-negative, symmetric weighting function w(t) to a truncated Fourier series in order to control the ringing (Gibbs phenomenum) introduced by the truncation [Kaiser,1966]. The ideal frequency response (Fig 2.5(b))

$$H(\omega) = \begin{cases} 1.0 , & -\omega_c \leq \omega \leq \omega_c \\ 0.0 & \text{otherwise} \end{cases} \qquad (2.18)$$

and $H(\omega) = H(-\omega)$

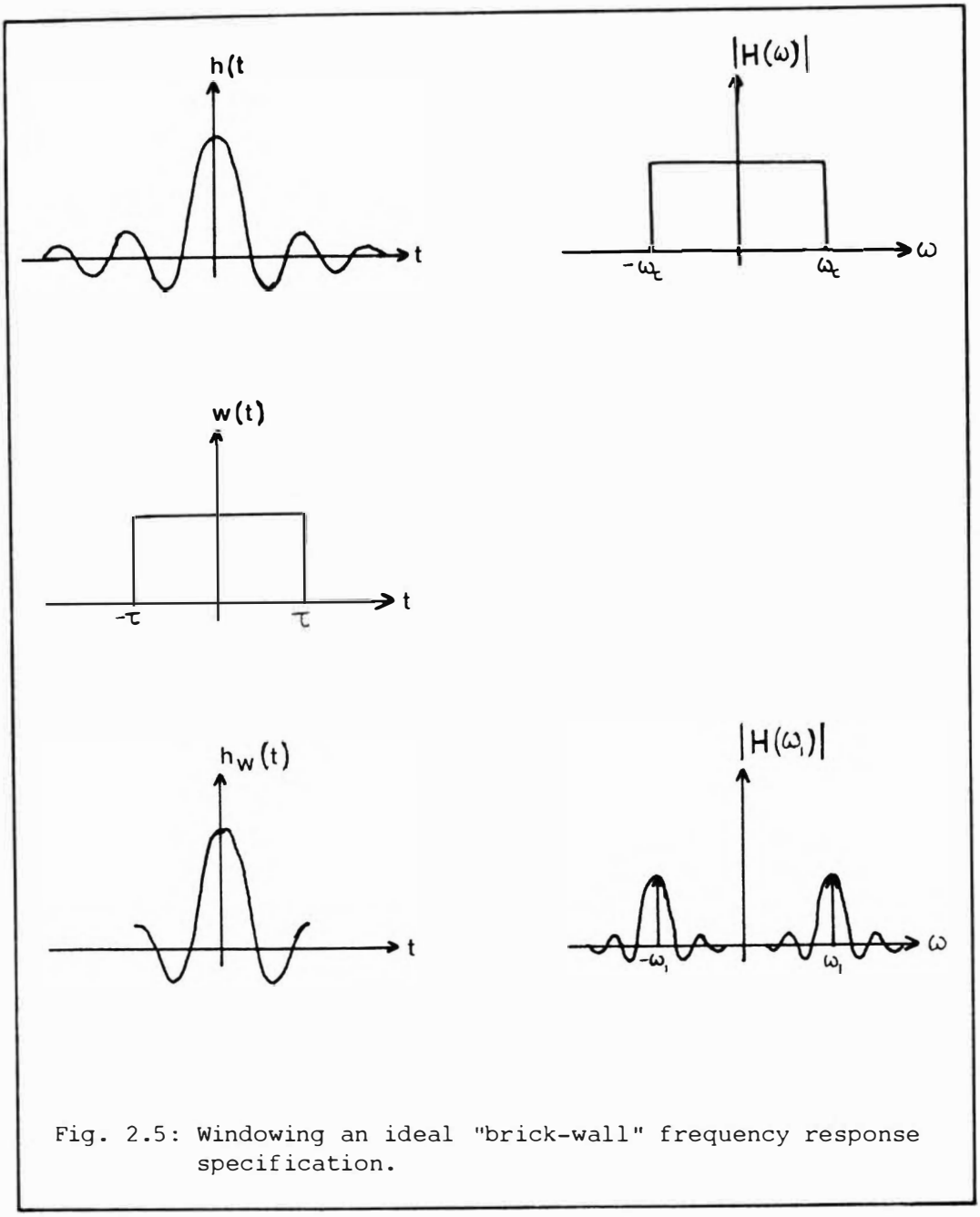


Fig. 2.5: Windowing an ideal "brick-wall" frequency response specification.

has, as its inverse Fourier transform, the impulse response sequence (Fig 2.5(a))

$$h(t) = \sin(\omega_c t)/\pi t \ , \qquad -\infty \leq t \leq \infty \tag{2.19}$$

If h(t) is time limited (truncated) by w(t) (Fig. 2.5(c))

$$w(t) = \begin{cases} 1.0 \ , & -\tau \leq \tau \\ 0.0 \ , & \text{otherwise} \end{cases} \tag{2.20}$$

$$w(t) = w(-t)$$

then the windowed function $h_w(t)$ (Fig. 2.5(d)) is

$$h_w(t) = h(t)w(t) \ , \qquad -\tau \leq t \leq \tau \tag{2.21a}$$

and has a Fourier transform

$$H_w(\omega) = H(\omega)*W(\omega) \tag{2.21b}$$

where * is the convolution operator and

$$W(\omega) = 2\sin(\omega T)/\omega \ , \qquad -\infty \leq \omega \leq \infty \tag{2.22}$$

Thus, examining any particular frequency ω_1 of $H(\omega)$, $-\omega_c \leq \omega_1 \leq \omega_c$ yields

$$H_w(\omega) = H(\omega_1)*2\sin(\omega T)/\omega$$

$$= 2H(\omega_1)\sin\frac{(\omega-\omega_1)T}{\omega-\omega_1} \tag{2.23}$$

as shown in Fig. 2.5(e). The application of the window has spread the single spectral component ω_1 over a region. The question then is what is the "best" window function w(t) that will minimise the width of the spreading and give low sidelobes in Fig 2.5(e). The short answer is there is no "best" window in an absolute sense. Harris [1978] has given a very comprehensive treatise on various aspects of many different windows.

However, at this point, knowing that the application of any window will inevitably introduce some spreading and have finite amplitude sidelobes, a more useful question to ask is what window will contain the maximum energy in the defined frequency response, relative to the total energy available. Define the ratio [Papoulis 1972]

$$\alpha = \frac{\int_{-\omega}^{\omega} |H(\omega)|^2 \, d\omega}{\int_{-\infty}^{\infty} |H(\omega)|^2 \, d\omega} \tag{2.24}$$

Then, the question becomes what is the function h(t), of finite extent 2τ, that maximises this energy ratio? The solution to this problem is known to be

$$h(t) = \begin{cases} \phi_0(t) \ , & |t| \leq \tau \\ 0 & \text{otherwise} \end{cases} \tag{2.25}$$

where $\phi_0(t)$ is the eigenfunction corresponding to the largest eigenvalue λ of

$$\int_{-\tau}^{\tau} \frac{\sin\omega_c(t-\mu)}{\pi(t-\mu)} \phi_n(t)\, d\mu = \lambda_n \phi_n(t) \qquad (2.26)$$

The solutions of equation (2.26) are known as the Prolate Spheroidal Wavefunctions of Slepian, Pollak and Landau [1961].

Out of all the possibilities of window functions w(t) that could be used, there are two windows that possess special properties. The first is the Kaiser window [Kaiser, 1966, 1974]. This window is a good approximation to the eigenfunction $\phi_0(t)$ of equation (2.25). the second window is the Dolph-Chebyshev window. This has the smallest side-lobe level for a given mainlobe width, or, conversly, the smallest mainlobe width for a given sidelobe level [Helms, 1968].

In its infancy, the process of windowing a Fourier series was laborious and relied on explicit analytic expressions being available to compute the Fourier series. Helms [1968] demonstrated how the FFT could be used to make the procedure very efficient by eliminating the need for explicit analytic formulas.

Very recently, Mathews et al [1985] describe the Discrete Prolate Spheroidal filter equivalent to equation (2.24) and compare it with, among others, the Kaiser and Dolph-Chebyshev window approach. Fig 2.6 is reprinted from their paper and shows the frequency responses of the different approaches. These plots are misleading because the authors adjusted the design parameters of the filters to give identical normalised 3 dB frequencies of 0.05. Such a practice is quite legitimate, but only serves to confuse the issue when trying to establish the differences between the windows for a fixed desired frequency response. In fact, of the three windows shown and for a fixed specification, the prolate spheroidal filter has the narrowest bandwidth (-3dB) point , but at the same time, the lowest sidelobe level.

2.3.3 Optimisation Techniques

All the remaining design techniques that will be discussed fall under this general heading. There are, however, many ways of "optimising" the design. With the exceptions of the methods involving explicit matrix inversion techniques, the common denominator in all of the optimisation techniques is the formulation of the problem in terms

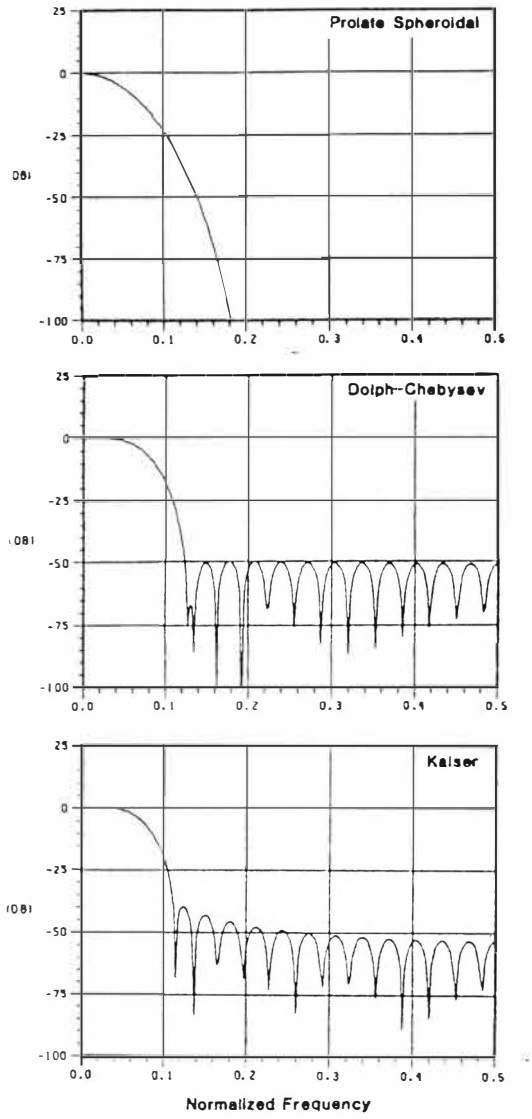


Fig. 2.6: Frequency response of the discrete prolate spheroidal filter compared with the Kiaser and Dolph-Chebyshev windows. [after Mathews et al, 1985]

of minimizing an error criterion by an iterative mathematical procedure. Filters designed by these methods are more ususally described as having equiripple or minimax responses. A minimax error [Helms, 1971] is one in which the minimum of the maximum absolute value of the error in the filter's response has been found. An equiripple filter possesses more than one maximum absolute value of this error and all of these errors are equal. A filter possessing one of these properties does not necessarily possess the other property. Some of the more important design methods in this category are

(a) frequency sampling technique

(b) Chebyshev approximation

(c) least square

2.3.3.1 Frequency Sampling technique

This method was originally proposed by Gold and Jordan [1969a] as a semi-automatic procedure, which was then made fully automatic by Rabiner et al [1970]. The principle of the method is to approximate the desired (continuous) frequency response with a finite number of samples N corresponding to the desired impulse response length and allowing a minimisation procedure to alter the magnitudes of (usually 2 or 3) of the frequency samples in the transition band in order to find the best fit. The procedure used by Rabiner [1970] was a steepest descent minimisation technique. Later Helms [1971] and Rabiner [1972] showed how the problem could be formulated as a constrained linear programming probem. This approach was more efficient and had fewer unconstrained variables. Convergence to the global minimum was also guaranteed.

McCreary [1972] has shown that one particular case of Rabiner's (Case C) is related to the direct Fourier series technique and amounts to choosing an optimum window. Steiglitz [1979] has shown that by adding suitable constraints, the linear programming approach will generate filters with monotone passband responses instead of minimax responses.

2.3.3.2 Chebyshev approximation

This method has received considerable attention in the literature and has proved very successful. The basic principle of the technique [Rabiner et al, 1975] is:

If $D(\omega)$ is the desired frequency response, $H(\omega)$ the actual frequency response of the filter whose impulse response coefficients are $h(n)$ and $W(\omega)$ is a non-negative weighting function on the approximation error in different frequency bands, form the weighted error function:

$$E(\omega) = W(\omega)\big(D(\omega)-H(\omega)\big) \qquad (2.27)$$

Then the Chebyshev approximation problem is to find the coefficients $h(n)$ that minimize the maximum absolute value of the error. ie

$$||E(\omega)|| = \underset{\text{coefficients}}{\text{minimise}} \left(\max_{\omega \in \Omega} |E(\omega)| \right)$$

where Ω represents the disjoint interval of the frequency bands of interest.

The method relies on the mathematical properties of Chebyshev functions [Rice, 1964a]. Parks and McLellan [1972] applied the Remez Exchange Algorithm [Rice, 1964b] which made the procedure efficient and guaranteed convergence. Later, the theory was unified to include all four cases of linear phase FIR filters [McLellan et al, 1973a], and their widely published algorithm [McLellan et al, 1973b] implements the theory. In recent years, several researchers have described acceleration procedures for the original algorithm to reduce the computation time significantly [Antoniou, 1983; Ebert et al, 1983]. In addition, the basic principle has been extended to give very flat passbands [Vaidyanathan, 1985] rather than equiripple passbands.

2.3.3.3 Least Square methods

The classical least-square approximation problem is to minimise the error function [Rice, 1964c]

$$e = \int_I (f(x)-\tilde{f}(x))^2 \, dx \qquad (2.28)$$

on an interval I, where $\tilde{f}(x)$ is the approximation to the desired function f(x). In terms of filter design, the problem can be considered from two distinct points of view.

The first approach is to consider the problem as a strictly mathematical optimisation problem: find the sequence h(n) that minimises the weighted error objective function

$$e = \sum_{k=1}^{M} (D(\omega_k) - H(\omega_k))^2 W(\omega_k)^2 \qquad (2.29)$$

where $H(\omega)$ is given by equation (2.4)
$W(\omega)$ is a non-negative symmetric weighting function

Lewis et al [1976] used a constrained linear programming approach which, when compared with a minimax method, gave improved stopband rejection and reduced passband ripple at the expense of a small increase in the transition width.

Cuthbert [1974a] used a numerical optimisation algorithm [Fletcher, 1971] with a weighting function $W(\omega)$ in the passband to give a

"crescent ripple" design. The algorithm used is specifically intended for sums-of-squares problems, but has considerable additional computational requirements. Re-writing equation (2.29) as

$$e = \sum_{k=1}^{M} r^2(k) \tag{2.30}$$

the algorithm requires the computation of the Jacobian matrix A and a vector V that is half the gradient vector [Cuthbert 1974a]:

$$A(i,j) = \sum_{k=1}^{M} (\partial r_k/\partial h_i)(\partial r_k/\partial h_j) \quad , \; i,j = 0,\ldots,(N/2)-1 \tag{2.31}$$

$$V(i) = \sum_{k=1}^{M} r_k(\partial r_k/\partial h_i) \quad , \; i = 0,\ldots,(N/2)-1 \tag{2.32}$$

Algazi and Suk [1975] showed that with a careful formulation of the problem in equation (2.29), with an arbitrary ideal lowpass specification

$$H_i(\omega) = 0, \quad |\omega| > \omega_c \tag{2.33a}$$

and frequency weight

$$|W(\omega)|^2 = \begin{cases} 1, & |\omega| < \omega_2 \;, \quad \omega_2 > \omega_c \\ A, & \text{otherwise} \end{cases} \tag{2.33b}$$

the solution reduces to finding the expansion coefficients h(k) of the solution h(t), where

$$h(t) = \sum_{k=1}^{\infty} h_k \phi_k(t) \quad , \qquad |t| \leq \tau \tag{2.34}$$

and the $\phi_k(t)$ are the prolate spheroidal functions of order k. Most importantly however, they demonstrated that the difficult problem of finding the explicit eigenvalues and eigenfunctions can be avoided and that practically, an iterative process

$$h_{k+1}(t) = D_\tau\left(\frac{1}{A}h_i(t) + \frac{A-1}{A}Bh_k(t)\right) \tag{2.35}$$

can be used, where $h_i(t)$ is the inverse Fourier transform of $H_i(\omega)$ and $h_{k+1}(t)$ is the (k+1)th approximation to the solution. If $h_k(t)$ has Fourier transform $G(\omega)$ and considering the term inside the () as g(t) then

$$D_\tau g(t) = \begin{cases} g(t), & |t| \leq \tau/2 \\ 0 & \text{otherwise} \end{cases}$$

$$BG(\omega) = \begin{cases} G(\omega), & |\omega| \leq \omega_2 \\ 0 & \text{otherwise} \end{cases} \qquad (2.36)$$

where ω_2 is defined in Fig. 2.3(b) and τ is the desired time length of the filter impulse response. The process successively bandlimits a previous approximation and then duration limits a weighted sum of the model and the bandlimited function $Bh_k(t)$. Comparing this approach with the earlier section on energy concentration, it is apparent from equation (2.34) that higher order eigenfunctions $\phi_k(t)$, $k > 0$ are included. The effect of this is to dramatically improve the behaviour of the passband response of Fig. 2.6. The beauty of the iterative process in equation (2.35) is that it only requires multiple applications of an FFT routine to implement it.

The second approach is statistical:
If the filter is considered as an estimator of a signal (the passband), with a noise component (the stopband), the problem is to find the weighting sequence (the impulse response h(n)) that gives the best (in a least-mean-square error sense) signal estimate and noise rejection. The classical solution is the Wiener filter [Bozic, 1979b].

The principle of the method is based on the well known convolution relation relating the input and output of a linear system: if the stationary, zero mean, random processes x(n) and y(n) are the input and output of a linear system with impulse response h(n), then

$$\tilde{y}(n) = \sum_{i=0}^{m} h(i)x(n-i) \qquad (2.37)$$

The estimation error is

$$e(n) = y(n) - \tilde{y}(n) \qquad (2.38)$$

where y(n) is the "desired" output sequence. The problem is then to minimise the mean-square error

$$\xi_N = E(|e(n)|^2) \qquad (2.39)$$

where $E(\ )$ is the expectation operator. The solution can be shown to that of finding the vector h(i) in the set of linear equations

$$R_N^{xx}\, h_N = R_N^{yx} \qquad (2.40)$$

where

$$R_N^{xx} = \begin{bmatrix} r^{xx}(1) & r^{xx}(2) & \cdots & r^{xx}(N) \\ r^{xx}(2) & r^{xx}(1) & \cdots & r^{xx}(N-1) \\ \vdots & & & \vdots \\ r^{xx}(N) & \cdots & & r^{xx}(1) \end{bmatrix} \qquad (2.41a)$$

$$R_N^{yx} = \begin{bmatrix} r^{yx}(1) \\ \vdots \\ r^{yx}(N) \end{bmatrix} \qquad (2.41b)$$

and $r^{xx}(k)=E[x(j+k)x^*(j)]$ is the autocorrelation function of the input $x(n)$ at lag k, and $r^{yx}(k)=E[y(j+k)x^*(k)]$ is the cross-correlation function between the output and input processes at lag k.

The solution of (2.40) requires the inversion of the matrix R_N^{xx}. For a linear phase FIR filter, this matrix is known to be a symmetric Toeplitz matrix. A Toeplitz matrix is one whose ij th element is a function only of i-j. Such a structure allows a drastic reduction in computational and memory requirements over standard matrix inversion routines [Farden, 1976]. One of the earliest fast methods for solving the system of equations in (2.40) is Levinson's Algorithm [Levinson, 1947]. Kellog [1972], Farden and Scharf [1974] and Manolakis et al (1982) have described this approach for the known statistics case. The major distinction between these approaches is that the method proposed by Farden and Scharf uses a weighted error formulation similar to that of Algazi and Suk. Subsequently, Clergeot and Scharf [1978] demonstrated the connection between the classical and statistical methods. More recently, Marple [1981] and Kalpouptsides et al [1985] have examined the more general unknown statistics case.

2.4 DESIGN METHODS IMPLEMENTED AND COMPARED

2.4.1 Existing Literature

Uniform comparison of the performance of alternative design methods from several points of view is fraught with difficulties because

(a) different methods have different sets of design parameters;

(b) in presenting examples of the performance of their methods, authors are not usually consistent in their choice of N or the

example;

(c) alternative ways of presenting the filter performance.

However, Tufts et al [1970] have compared the prolate-spheroidal filters of Slepian et al and the least-mean-square method of Rorabacher against a minimax design. They used a fixed sequence length N=11 and compared the relative performance of each method in relation to the "desired" ω_p and ω_s of Fig 2.3(a). In contrast, Rabiner [1971] compares the window, frequency sampling and equiripple procedures from the point of view of normalised transition width D versus peak stopband ripple δ_s with the peak passband ripple δ_p as a parameter. He defined the normalised transition bandwidth D as

$$D = NT(\omega_s - \omega_p)/2\pi \qquad (2.42)$$

Fig 2.7 is reproduced from his paper. His results show that the equiripple methods (Chebyshev approximation) are optimal in minimising the transition width for a fixed sequence length N.

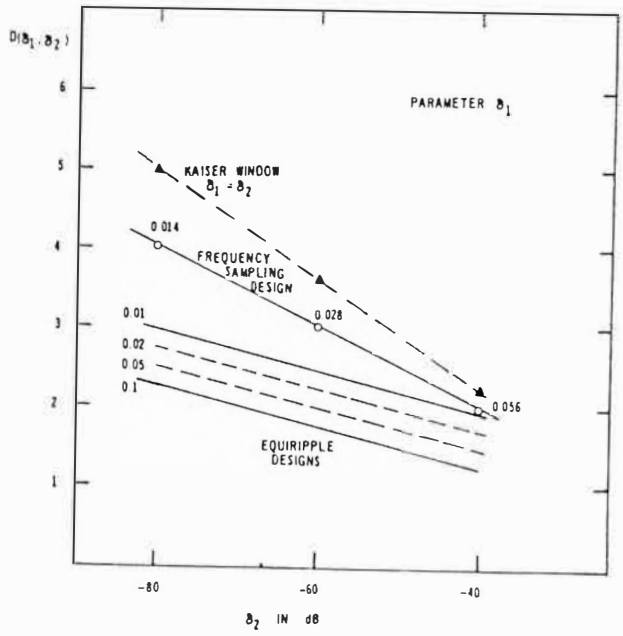


Fig. 2.7: Normalised transition bandwidth versus stopband ripple for three different design methods. [after Rabiner, 1971]

Herrmann et al [1973a] in a comprehensive paper discussed practical design rules for lowpass filters using Chebyshev minimax methods. They demonstrated the relationship between the parameters as

$$D = D_\infty(\delta_p,\delta_s) - f(\delta_p,\delta_s)(\Delta F)^2 \qquad (2.43)$$

where $D = (N-1)\Delta F$

$\Delta F = (\omega_s - \omega_p)/2\pi$

$K = \delta_p/\delta_s$

$$D_\infty(\delta_p,\delta_s) = [a_1(\log_{10}\delta_p)^2 + a_2\log_{10}\delta_p + a_3] + [a_4(\log_{10}\delta_p)^2 + a_5\log_{10}\delta_p + a_6]$$

$$f(\delta_p,\delta_s) = b_1 + b_2\log_{10}\delta_p - b_2\log_{10}\delta_s$$

$a_1 = 5.309\times10^{-3}$ $\quad$ $b_1 = 11.01217$

$a_2 = 7.114\times10^{-2}$ $\quad$ $b_2 = 0.51244$

$a_3 = -4.761\times10^{-1}$

$a_4 = -2.66\times10^{-3}$

$a_5 = -5.94\times10^{-1}$

$a_6 = -4.278\times10^{-1}$

Subsequently, Rabiner [1973] showed how this relationship could be used with the McLellan-Parks program so that given any 4 of the five parameters N, ω_p, ω_s, δ_p, δ_s, the program will give the optimal Chebyshev solution. This is the only case for which the relationship between the basic design parameters has been determined.

The prolate spheroidal filters of Mathews et al [1985], and Tufts et al [1970], are the zeroth order eigenfunction corresponding to the largest eigenvalue of the characteristic equation (2.26). A major penalty of the method is a large distortion of the passband while giving extremely low sidelobes. In addition, the transition width is very much wider than solutions derived from minimax methods. Using Algazi and Suk's method to include higher order eigenfunctions improves matters considerably and gives solutions that are comparable with those provided by Chebyshev methods. Fig. 2.8 is reproduced from their paper. In addition, they demonstrate the relationship between the transition bandwidth and their weight parameter A of equation (2.35) in a similar manner to Herrmann et al. Fig. 2.9 is also reproduced from the paper of Algazi and Suk.

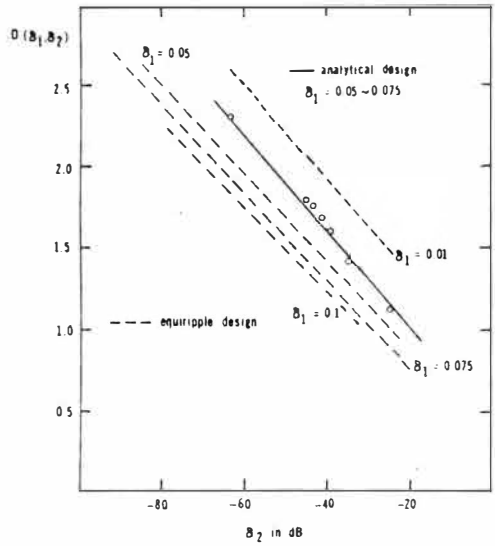


Fig. 2.8: Algazi's algorithm compared with equiripple methods.
[after Algazi et al, 1976]

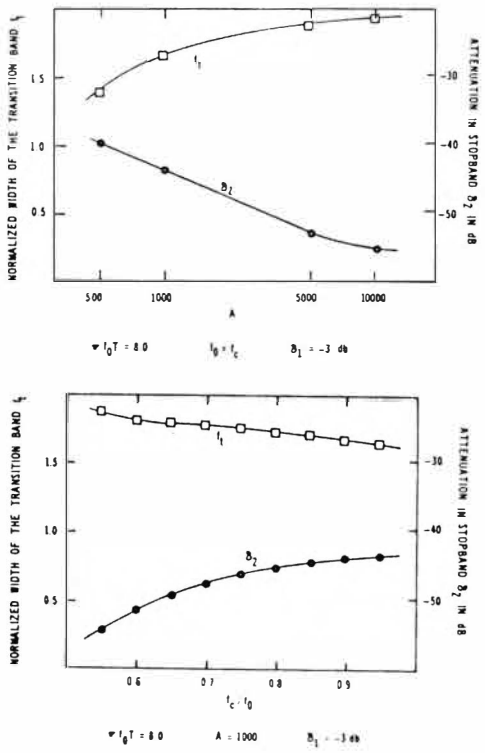


Fig. 2.9: Effects of A and f_c on the normalised transition bandwidth and stopband ripple. Solution is via equation 2.34. f_c corresponds to ω_c of the text; f_o is related to the weight function magnitude - see fig. 2.10(a)

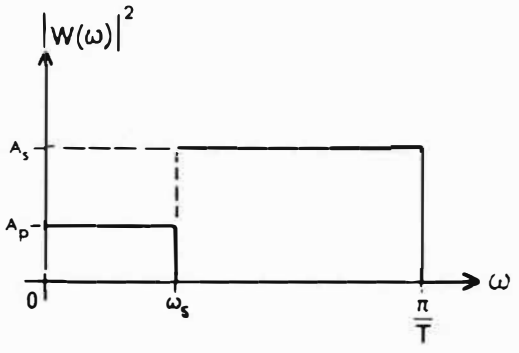
Other writers - such as Farden et al [1974] present their information in a manner that makes it almost impossible to gain a meaningful comparison. As a final point before discussing a common basis for comparison, there is very little reported information on the algorithmic efficiency of different algorithms from the joint perspective of code complexity and elapsed time to produce a solution. Some methods - such as the McLellan-Parks algorithm are very elegant, but, in the absence of the published program, are very complex to code. The window methods only require an FFT routine, but the mathematical optimisation methods can have very large storage requirements and considerably longer execution times to provide a solution.

2.4.2 A common comparison base

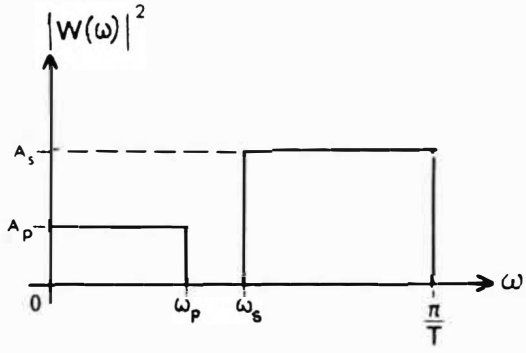
The previous section showed that while some comparisons have been reported, their effectiveness in informing the potential users is entirely dependent on the knowledge and familiarity of the users with the details of the design methods.

A more useful approach described here, is to define a common "ideal" filter and compare the results of the different methods using a common set of defined parameters. Three types of comparison will be made:

(a) frequency response
(b) CPU time for solution
(c) data storage requirements



(a) common base adopted excluding McLellan-Parks algorithm



(b) weight specification for for McLellan-Parks only.

Fig. 2.10: Weighting function and A_p, A_s defined.

The FFT routine FFT842 [Bergland et al, 1979] is the common denominator used in establishing the frequency response of the filters. This FFT routine explicitly retains the real and quadrature components over the whole of the first Nyquist interval (ie 0 - Fs, Fs the sampling frequency). For linear phase filters, most of this information is redundant. However, in the work on non-linear phase filters to follow, all the information is required by some of the design methods. The principal frequency domain parameters to be used were defined in section 2.2.2 and shown in Fig. 2.3(b). Such a specification allows the McLellan-Parks program to be consistent with the other methods by specifying only ω_p and ω_s and ignoring ω_c .

Some of the algorithms require a weight function $W(\omega)$, and different authors have used slightly different variations. In this work, the procedure of Algazi and Suk is adopted in defining $W(\omega)$. This is shown in Fig. 2.10(a) for a lowpass filter. The reader is cautioned not to confuse this weight specification of Fig 2.10(a) (for the computer algorithm) with the filter specification of Fig. 2.3(b). The single exception to this rule is the McLellan-Parks algorithm that requires $W(\omega)$ to be specified on the same disjoint frequency interval as the desired function. This is shown in Fig. 2.10(b) for the same lowpass filter.

The methodology adopted in producing the comparisons is shown in Fig. 2.11. The McLellan-Parks algorithm source code from the IEEE package was modified slightly to separate the input and output blocks as separate subroutine calls and to include the VAX system specific CPU timer calls. The output code was altered to allow the result to be written to a disk file which could be read by the main program and thus use the common graphics package.

The design methods chosen were

1. Windows - Kaiser
   - Dolph-Chebyshev
2. McLellan-Parks algorithm (Chebyshev approximation)
3. Cuthbert (1974a) (least square, numerical optimisation)
4. Algazi-Suk algorithm (iterative procedure)
5. Clergoet-Scharf (1978) (statistical formulation, Toeplitz matrix inversion)

The choice was motivated by the desire to minimise the overall computational effort involved in getting the algorithms running, the availability of software such as the Toeplitz matrix inversion

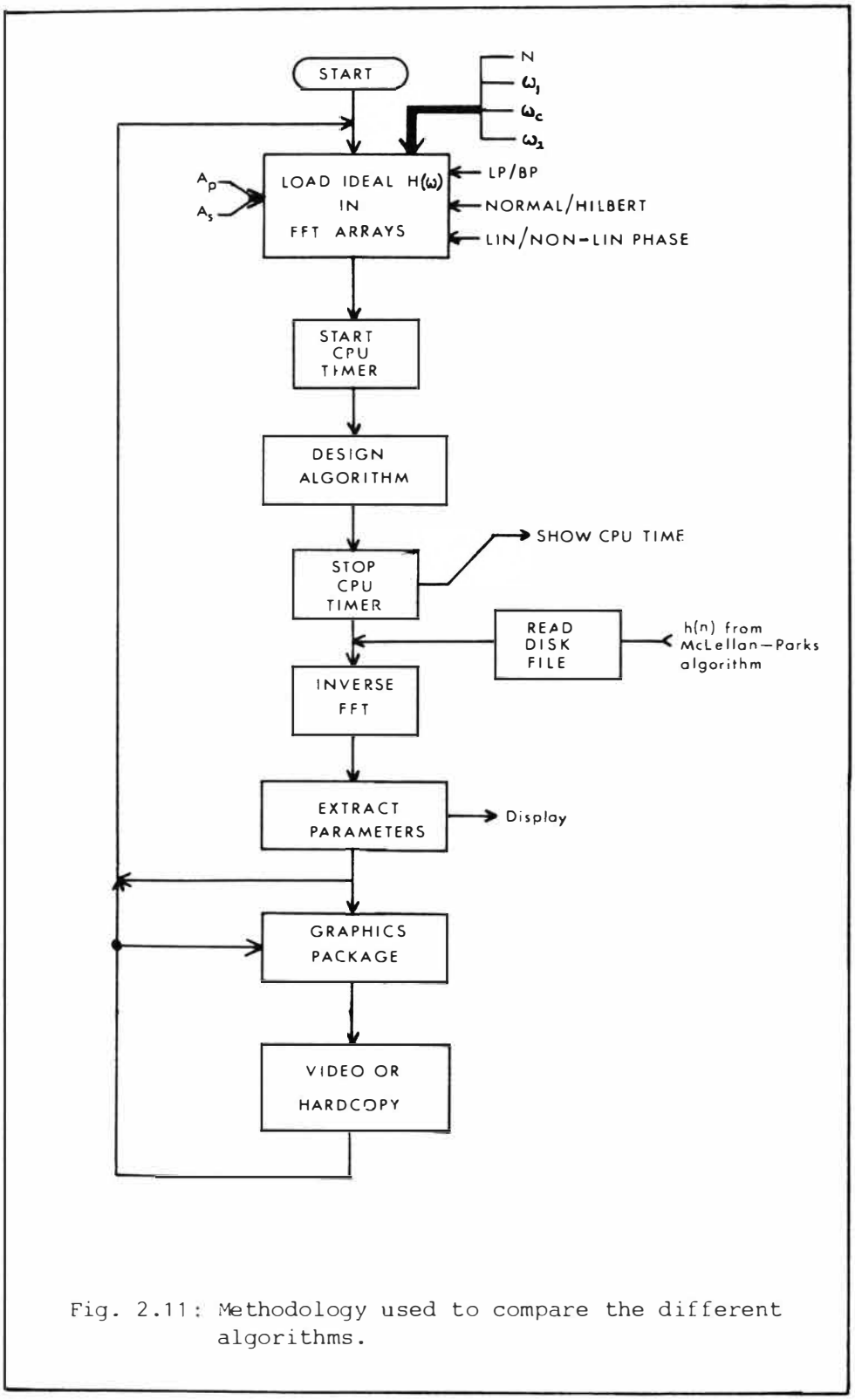


Fig. 2.11: Methodology used to compare the different algorithms.

routine, and the potential of the method to be extended to include the non-linear phase case to follow.

For Cuthbert's method, his original weighting function in the passband giving cresent ripple was substituted by that defined above. In addition, he used the subroutine E04GAF from the NAG (Numerical Algorithms Group) Library. This particular routine was not available (on the VAX), but an equivalent subroutine VA07A from the Harwell Library was available and used the same algorithm by Fletcher.

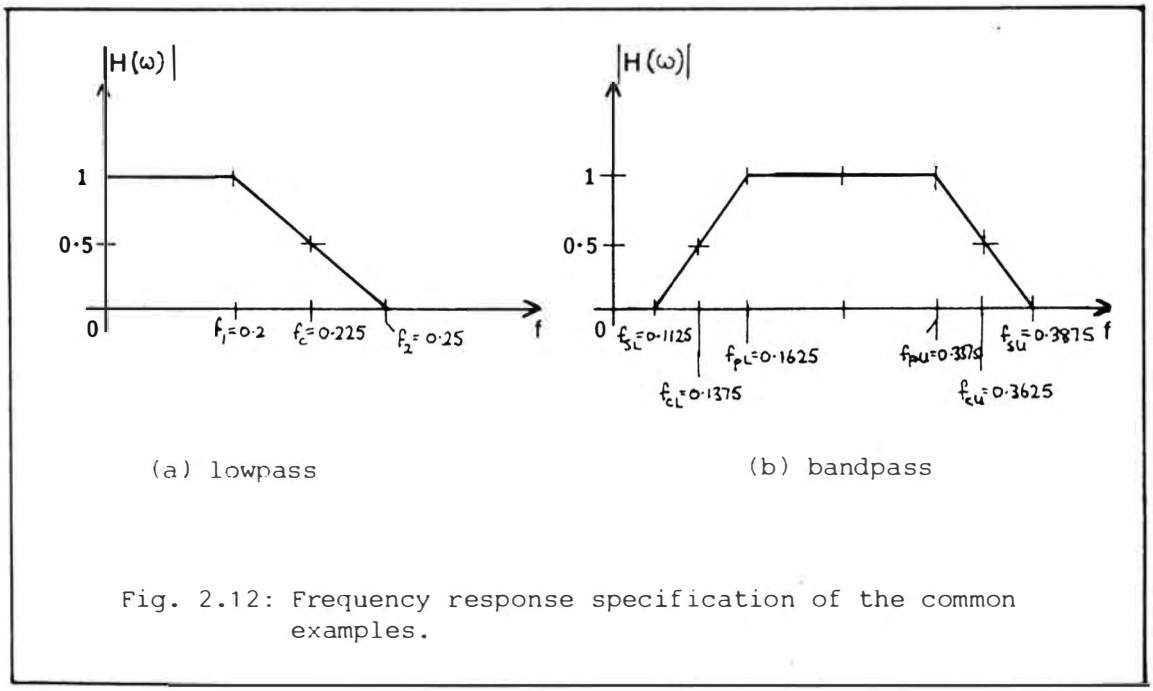
For the Statistical formulation, the Fortran subroutine TPLTZ of Farden [Farden et al, 1974], and reprinted later in Farden [1975] was used as the Toeplitz inversion routine.

All the calculations and results are based on single precision floating point representation.

Two examples will be used: a lowpass and bandpass filter. The bandpass example is a frequency translated version of the lowpass example and retains the same bandwidth (between -6dB points) and transition widths. The parameters of the normlised "ideal" frequency responses used as the common examples are shown in Fig. 2.12.

2.4.3 Lowpass Filter example

The next step is to choose a filter length N. For the moment, consider only the lowpass example: the choice of filter length N, transition width and stopband ripple are not independent. For



(a) lowpass (b) bandpass

Fig. 2.12: Frequency response specification of the common examples.

Algazi-Suk, they are constrained by the time-bandwidth product

$$C = \pi N \Delta F \qquad (2.44)$$

where $\Delta F = (\omega_s - \omega_p)/2\pi$

N = number of impulse response coefficients desired

In addition, both the Kaiser and Dolph-Chebyshev windows are parameterised on α and N [Kaiser,1974; Harris,1978, equation (46a)] which is a different way of indicating time-bandwidth dependence. Selecting, arbitrarily, an acceptable sidelobe level of -60 dB, ΔF=0.05 with a normalised sampling interval T=1, Algazi and Suk have tabulated the minimum stopband attenuation as a function of C. Thus from (2.44), N=51. For the Chebyshev solution, Rabiner's (1973) relationships of equation (2.43) also suggest N=51 for K=10. However, Kaiser's empirical relationships

$$\alpha = \begin{cases} 0.1102(\text{ATTEN}-8.7) & \text{ATTEN}>50 \\ 0.5842(\text{ATTEN}-21)^{0.4}+0.07886(\text{ATTEN}-21) & 21\leq\text{ATTEN}\leq 50 \\ 0.0 & \text{ATTEN}<21 \end{cases} \qquad (2.45a)$$

$$N-1 \approx \frac{\text{ATTEN}-8}{14.36\Delta F} \qquad (2.45b)$$

where ATTEN is the required attenuation in dB as a positive number, ΔF is the normalised transition width, suggest that α=5.653 and N=73, so the larger value of N=73 is used here.

Table I and Fig. 2.13 present the frequency responses of the five methods. Table II is the CPU time required for a solution, and Table III indicates the amount of extra array storage beyond the basic FFT arrays, required by each algorithm. The CPU times reflect fully optimised compiled code with no VAX DEBUG code included.

The results for the window methods using the example of Fig. 2.12(a) are misleading because a transition band has been explicitly included. The principle of the window method is that it is intended to be used in conjunction with an ideal "brick-wall" specification that corresponds to Fig 2.4(b). Thus, using

$$|H(f)| = \begin{cases} 1 & , |f| < 0.225 \\ 1/2 & , |f| = 0.225 \\ 0 & , |f| > 0.225 \end{cases}$$

$$f = \omega T/2\pi, \quad T=1$$

TABLE I: COMPARISON OF THE RESULTANT FREQUENCY RESPONSE OF 5 DESIGN ALGORITHMS: LOWPASS FILTER

normalised frequency $f = \omega T/2\pi$, T=1
impulse response length N = 73
$dB = 20.0*LOG_{10}(ratio)$

Algorithm	f_p	f_c	f_s	parameter	δ_p (dB)	δ_s (dB)
Desired	f_1=0.20	0.225	f_2=0.250			< -60
(i) Windows:						
Kaiser	0.178	0.225	0.274	-	0.00035	-86.0
Dolph-Cheb	0.171	0.225	0.282	-	-0.0025	-69.7
(ii) McLe-Parks	0.200	0.222	0.252	NGRID=16 K=10	0.015	-75.5
(iii) Cuthbert	0.198	0.225	0.253	Ap=1 As=100 iter=3	0.247	-61.28
(iv) Algazi-Suk	0.199	0.225	0.254	A = 10^5 iter=10	0.114	-44.25
(v) Toeplitz	0.002 see plot	0.172	0.250	Ap=1.0 As=10^5	-0.203 see plot	-62.07

TABLE II: CPU TIME REQUIRED FOR SOLUTION: LOWPASS FILTER

Algorithm	time (s)	Parameter
(i) Windows:		
Kaiser	0.32	FFT len=1024
Dolph-Cheb	0.84	
(ii) McLe-Parks	9.39	NGRID=16 K=10
(iii) Cuthbert	52.77	order 73x512
(iv) Algazi-Suk	5.81	FFT len=1024
(v) Toeplitz	0.46	order 73x73

TABLE III: PRINCIPAL ARRAY STORAGE REQUIREMENTS OF THE 5 ALGORITHMS IN EXCESS OF THE BASIC FFT ARRAYS USED TO DEFINE THE DESIRED FREQUENCY RESPONSE.

N is the desired impulse response length
M is the number of frequency points used to specify the frequency domain information
NFFT is the FFT length used

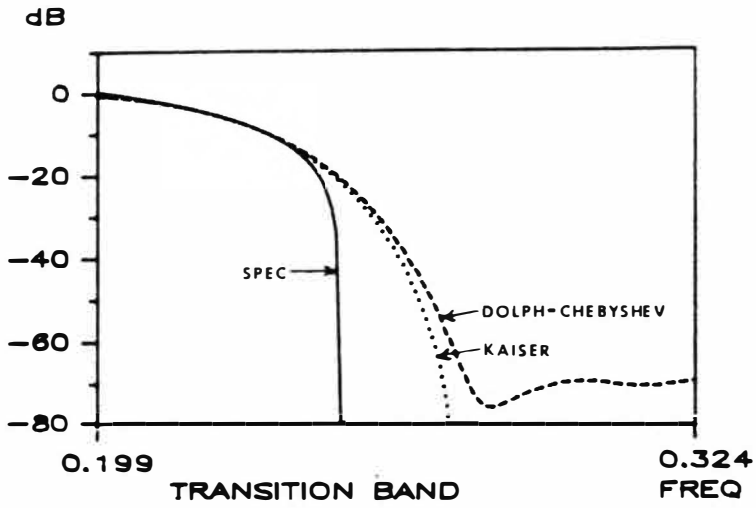
symbology is PxL: no of arrays P, of length L required.

Algorithm	Extra array storage
(i) Windows:	
Kaiser	0
Dolph-Cheb	1xNFFT
(ii)	
McLe-Parks (N≤128)	1x20; 3x10; 6x66; 3x1056
(iii)	
Cuthbert	2xNFFT; 3xM; 7xN; 1 x (NxN); 1 x (MxN)
(iv)	
Algazi-Suk	1xNFFT
(v)	
Toeplitz	4xN; 1x([N+1]**2/4)

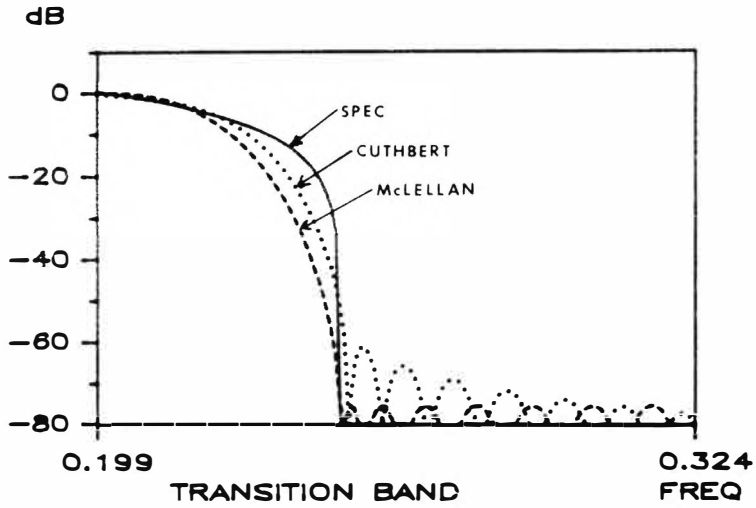
TABLE IV: FREQUENCY RESPONSE PARAMETERS OF WINDOW METHODS WITH A "BRICK-WALL" DESIRED RESPONSE. (LOWPASS FILTER)

normalised frequency $f = \omega T/2\pi$, T=1
impulse response length N = 73
dB = $20.0*LOG_{10}$(ratio)

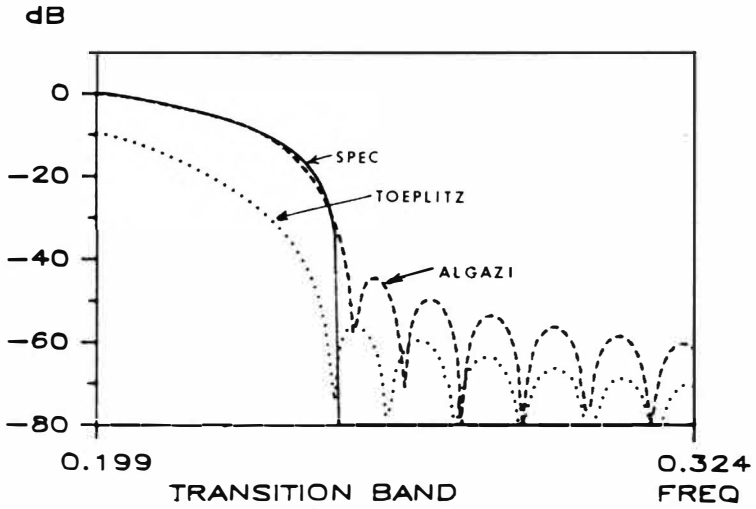
Algorithm	f_p	f_c	f_s	parameter	δ_p (dB)	δ_s (dB)
Desired	-	0.225	-			< -60
(i) Windows:						
Kaiser	0.201	0.225	0.251	-	0.009	-61.2
Dolph-Cheb	0.186	0.225	0.264	-	0.0004	-62.4



(a) transition band region for the windows.

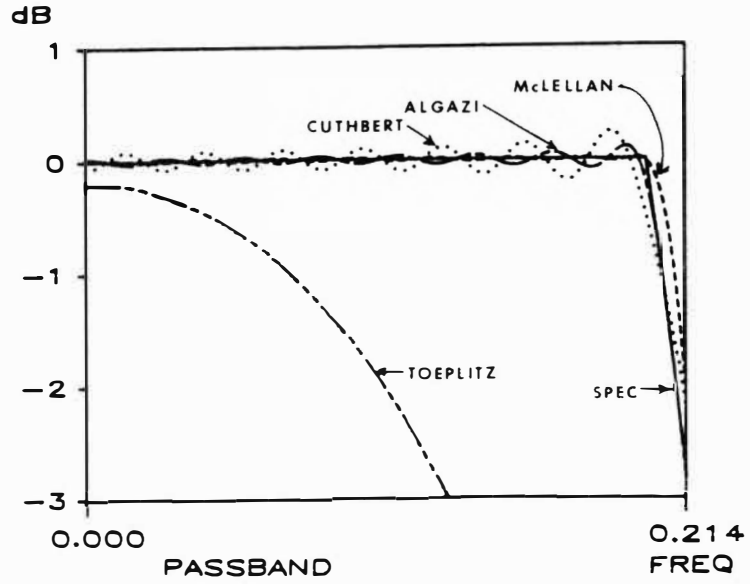


(b) transition band region for McLellan-Parks and Cuthbert's methods.



(c) transition band region for Algazi's and Toeplitz methods.

Fig. 2.13: Frequency responses achieved by the different design methods.



(d) passband response for the optimisation methods.

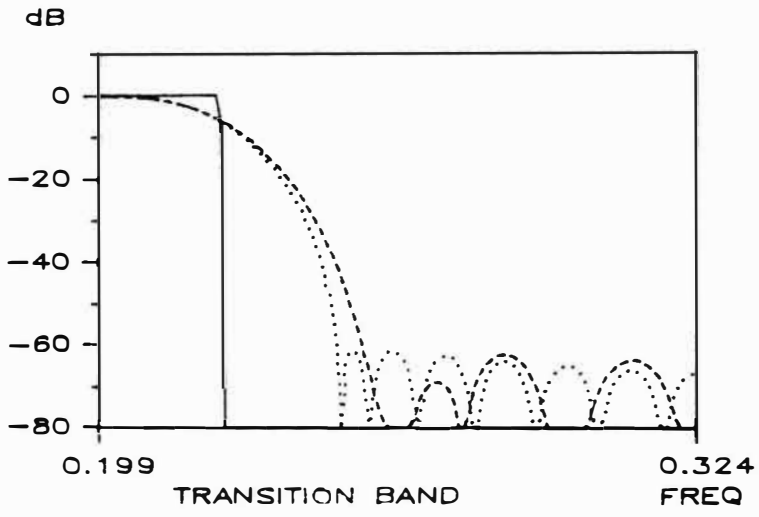


Fig. 2.14: Window methods applied to the ideal "brick-wall" specification

and applying the window techniques only, Table IV and Fig 2.14 indicates the results. Fig 2.14 should be compared with Fig 2.13(b) and (c). The stopband ripple is within the required tolerance and the total width of the filters as given by f_c in Tables I and IV are very similar.

2.4.4 Bandpass Filter Example

In exactly the same manner used for the Lowpass example above, Fig. 2.15 and Table V present the frequency response parameters for the bandpass specification of Fig. 2.12(b). Fig 2.16 and Table VI are the result for the window methods applied to the equivalent "brick-wall" band-pass specification and should be compared with Fig. 2.15(b),(c) and (d). For this example, there are two sets of paramaters corresponding to the lower and upper transition bands. The suffix 'L' or 'U' identify the appropriate band. The plots of Fig. 2.15 and 2.16 show the upper transition band.

2.4.5 Discussion

The key conclusions to be drawn from Table I,IV and V,VI are:

(a) with the exception of method (iv), all techniques exceed the required stopband ripple specification.

(b) for the lowpass example, methods (i), (iii) and (iv) preserve the specified cutoff frequency, while that for method (ii) represents a 1.3% shift. This preservation of the -6 dB frequency is the principal reason for frequency specification adopted in Fig. 2.3(b). That the window methods preserve the -6dB point is well known. However, such behaviour of the McLellan-Parks algorithm (method (ii)) has not been reported, because the algorithm prohibits any specification or control over the transition region.

For the bandpass example, the identical comments apply to the upper -6dB frequency f_{cU}. However, for the lower -6dB frequency f_{cL}, only methods (i) and (iv) preserve the specification. Method (ii) gives a +2.25% shift and method (iii) a +0.8% shift.

(c) the transition bandwidth is maintained for all the methods except for the Dolph-Chebyshev window (Table IV).

(d) The statistical formulation of method (v) produces a very poor result in every respect except the stopband ripple requirement.

TABLE V: COMPARISON OF THE RESULTANT FREQUENCY RESPONSE OF 5 DESIGN ALGORITHMS: BANDPASS FILTER

normalised frequency $f = \omega T/2\pi$, $T=1$
impulse response length $N = 73$
$dB = 20.0*LOG$ (ratio)

Algorithm	f_{sL}	f_{cL}	f_{pL}	f_{pU}	f_{cU}	f_{sU}	parameter	δ_p (dB)	δ_{sL} (dB)
Desired	0.1125	0.1375	0.1625	0.3375	0.3625	0.3875			< 60.
(i) Windows:									
Kaiser	0.0879	0.1377	0.1846	0.3154	0.3623	0.4121	} NFFT=1024	0.00046	-85.0
Dolph-Cheb	0.082	0.1377	0.1904	0.3096	0.3623	0.4180	}	0.00012	-90.0
(ii) McLe-Parks	0.1114	0.1406	0.1621	0.3379	0.3623	0.4180	K=10 NGRID=16	0.013	-76.1
(iii) Cuthbert	0.1104	0.1387	0.1650	0.3359	0.3623	0.3906	Ap=1 As=100 iter=3	0.2.96	-61.9
(iv) Algazi-Suk	0.1084	0.1377	0.1631	0.3369	0.3623	0.3916	A = 10 iter=10	0.111	-44.1
(v) Toeplitz	0.1093 see plot	0.1621	0.1631	0.3369	0.3379	0.3906	Ap=1 As=10	0.253	-64.0

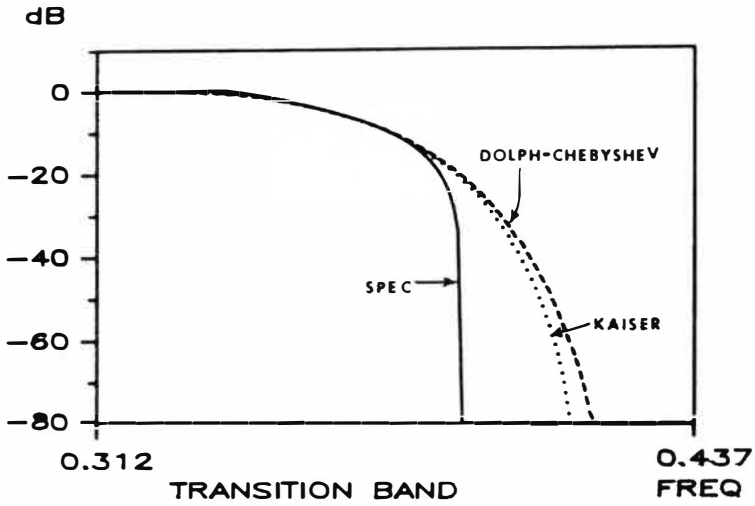
TABLE VI: FREQUENCY RESPONSE PARAMETERS OF WINDOW METHODS WITH A "BRICK-WALL" DESIRED RESPONSE: BANDPASS FILTER

normalised frequency $f = \omega T/2\pi$, $T=1$

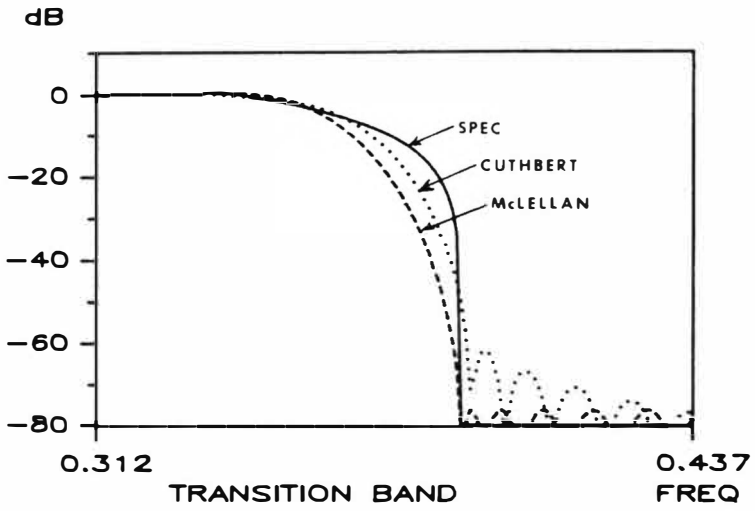
impulse response length $N = 73$

$dB = 20.0*LOG_{10}(ratio)$

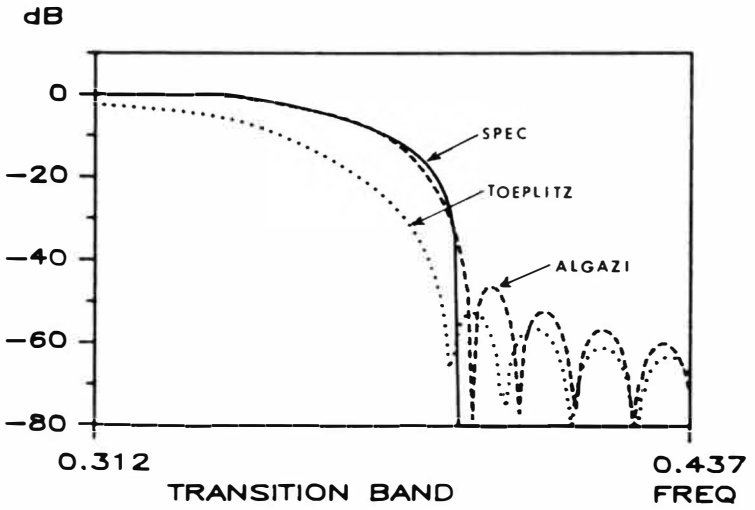
Algorithm	f_{sL}	f_{cL}	f_{pL}	f_{pU}	f_{cU}	f_{sU}	parameter	δ_p (dB)	δ_{sL} (dB)
Desired	-	0.1375	-	-	0.3625	-			< 60.
(i) Windows:									
Kaiser	0.1113	0.1377	0.1621	0.3379	0.3623	0.3887	} NFFT=1024	0.0085	-61.1
Dolph-Cheb	0.0986	0.1377	0.1689	0.3311	0.3623	0.4014	}	0.0023	-68.5



(a) transition band region for the windows

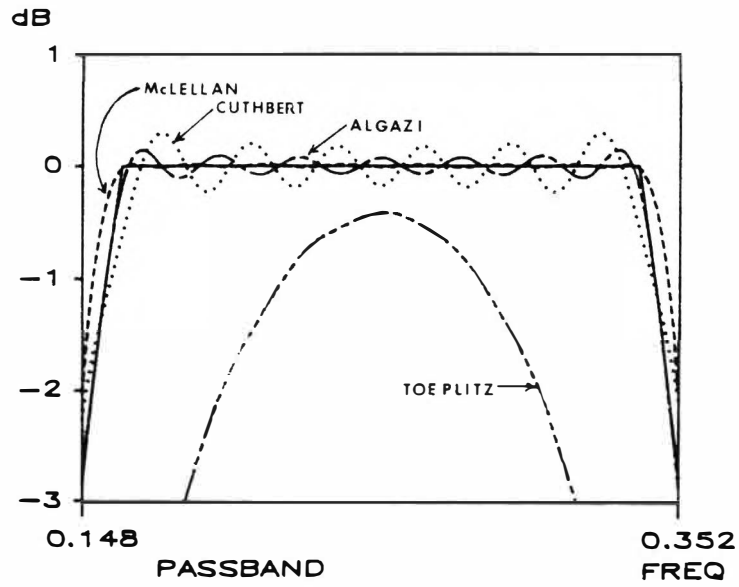


(b) transition band region for McLellan-Parks and Cuthbert's optimisation methods.

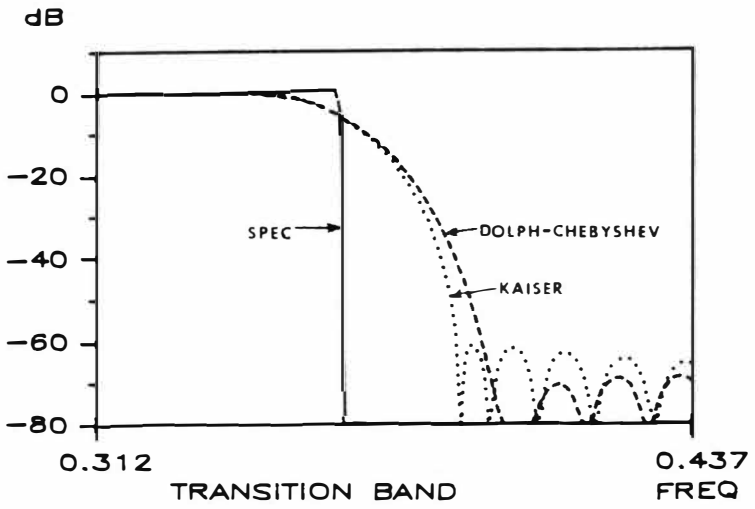


(c) transition band region for Algazi's and Toeplitz methods.

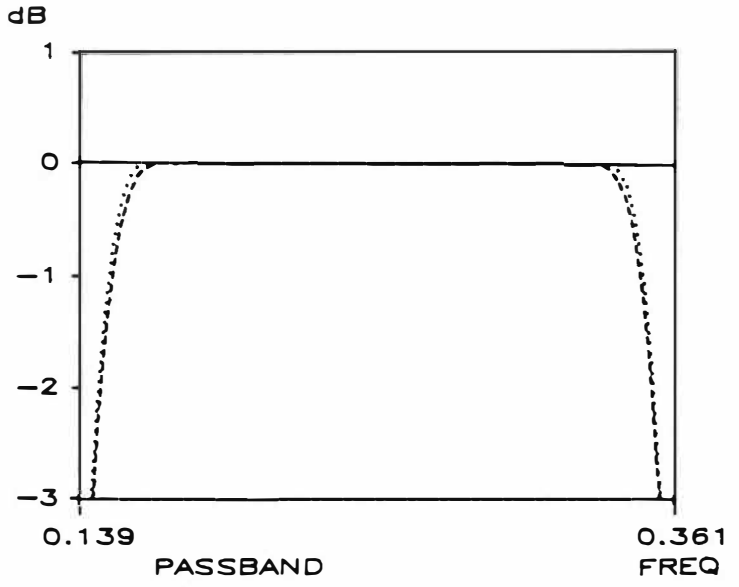
Fig. 2.15: Bandpass Example - frequency responses achieved by the different design methods.



(d) passband response for the optimisation methods.



(a) transition band region: compare with Fig. 2.15(b),(c)



(b) passband response. Compare with Fig. 2.15d.

Fig.2.16: Bandpass example: window methods applied to the "brick-wall" ideal specification.

With respect to the CPU time required for a solution, all the methods except Cuthbert's give fast solution times. This makes them useful in an experimental environment where different filters and combinations of parameters can be synthesised and compared rapidly.

Cuthbert's method by comparison is 50-100 times slower. This is due to the large amount of computation required by the numerical procedure. At each iteration, equations (2.31) and (2.32) must be computed and the solution of

$$\sum_{j=1}^{n} a_{ij} x_j = b_i \qquad i = 1,....n \tag{2.46}$$

must be found. The solution of equation (2.46) is done by traditional matrix inversion techniques which are considerably slower than a solution that can exploit a Toeplitz structure.

From the point of view of computer storage requirements, the most efficient algorithms are those of methods (i) and (iv), followed by McLellan-Parks (ii), and then Clergeot-Scharf (v). Cuthbert's method requires a very large amount of array storage reflecting the embedded matrix computations mentioned above.

More will be said about Cuthbert's method in Chapter 3, but Appendix 1 gives the computational details relating to actually implementing the algorithm. However, from Appendix 1, for a fixed choice of N, the amount of storage and CPU time required is controlled by M: the number of frequency points used to specify the model or desired frequency response.

In the results reported above, M=512 corresponding to the Nyquist interval of the frequency specification setup in the FFT arrays. Thus M also controls the accuracy of the frequency specification used by the optimisation procedure. Hence, if accurate specification of the cut-off frequency ω_c and transition bandwidth is not required, M can be reduced to smaller values with a consequent reduction in array storage and faster solution times.

2.5 THREE ALGORITHMS REVISITED

At this point, the algorithms of McLellan-Parks, Algazi-Suk and Cuthbert are revisited. For the McLellan-Parks algorithm, certain

shortcomings (connected with filters requiring odd symmetric sequences) are pointed out, as they have a bearing on the discussion of non-linear phase filters that is to follow. For the algorithm of Algazi-Suk, some practical aspects related to the value of their weight parameter A are examined, and the use of the iterative procedure to "refine" the result derived from method 5 (Toeplitz solution). For Cuthbert's method, aspects related to the weight parameters A_p and A_s are examined.

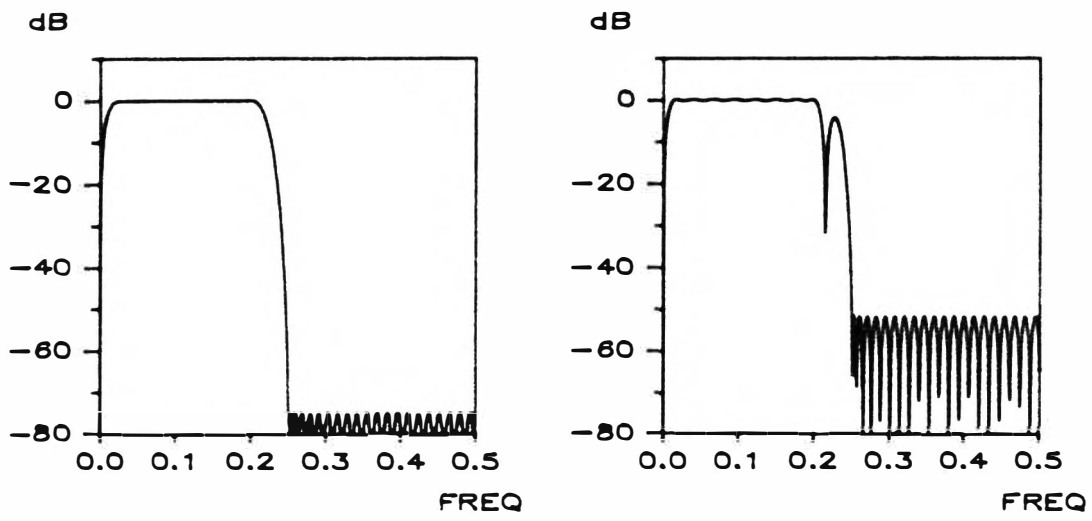
2.5.1 McLellan-Parks algorithm

For ordinary Case 1 or Case 3 (equations 2.17a,c) lowpass filters, the specification of the desired passband explicitly includes the $\omega=0$ point. For the common example of section 2.4.2, this is (0.0,0.2,0.25,0.5).

The discussion on non-linear phase filters in Chapter 3 will show that anti-symmetric sequences of Case 2 or Case 4 will be required. However, a problem arises in specifying $\omega=0$ for the equivalent Hilbert filter. Hilbert filters are represented by Case 2 or Case 4 (equation 2.17b,d) and such sequences require $|H(\omega=0)|=0.0$ from equation 2.13. Thus, a passband specification $|H(\omega=0)| > 0$ is illegal.

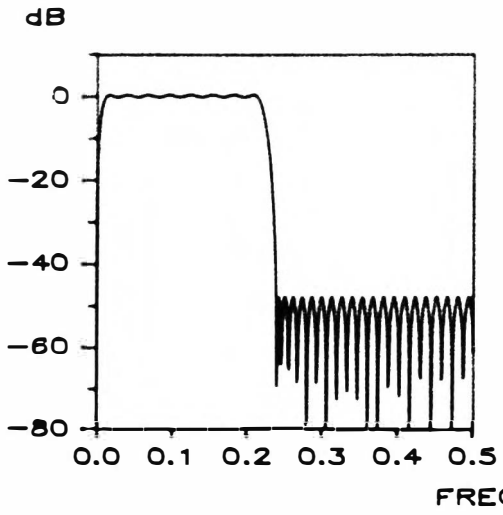
The question that arises then, is how close can the lower passband specification get for a fixed choice of N without incurring a major deterioration in the resulting frequency response? In their original paper describing the program, McLellan et al gave an example [McLellan, 1973b, Fig. 19,20] with a lowest (normalised) band edge of 0.05. If the sampling rate of a speech processing system is, say, 8 KHz, this bandedge is 0.05 x 8 KHz = 400 Hz. The speech processing of Chapter 4 will indicate that passband responses down to at least 50 Hz (at an 8 KHz sampling rate) will be required. Normalising 50 Hz to T=1, gives a required lower passband edge of 0.00625.

Experience with this algorithm has shown that for Hilbert filters, the allowable lower passband edge specification and transition band interval are not independent of each other. Fig 2.17 demonstrates the interaction obtained. The starting point is the lowpass common example of Fig. 2.12(a) with the lowest passband edge lifted (arbitrarily) to 0.025. Fig. 2.17(b) then halves this passband edge to 0.0125 while maintaining the same transition width. The result is that the upper region of the passband deteriorates significantly. Reducing the

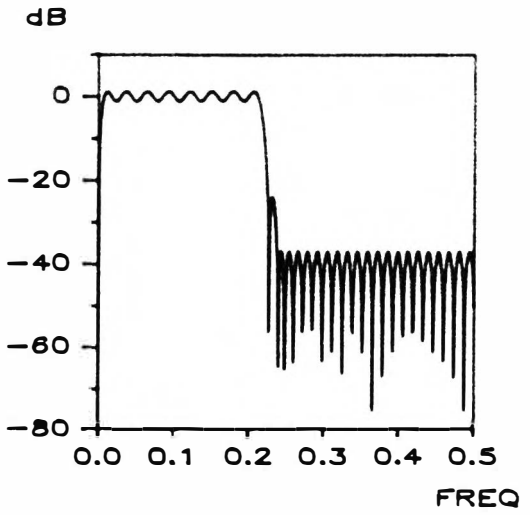


(a) spec = (0.025,0.2,0.25,0.5)

(b) spec=(0.0125,0.2,0.25,0.5)



(c) spec=(0.0125,0.2125,0.2375,0.5)



(d) spec=(0.00625,0.2125,0.2375,0.5)

Fig. 2.17: Hilbert Filter via McLellan-Parks algorithm: demonstration of the interaction between the lowest passband frequency specification and transition bandwidth specification.

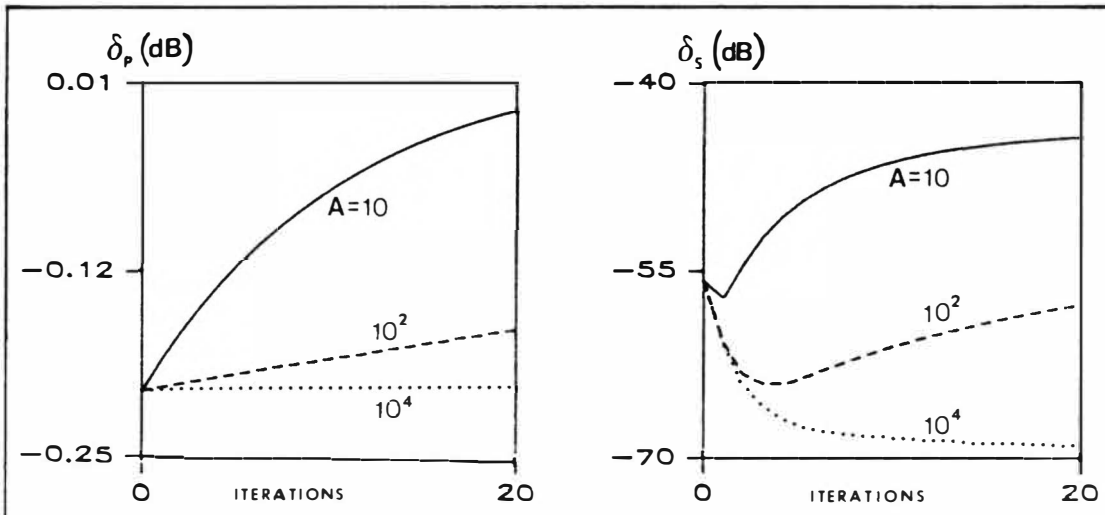
transition bandwidth by half symmetrically about f_c without altering the new lower passband edge restores the passband (Fig 2.17(c)) but at the expense of greatly increased passband and stopband ripple because the transition bandwidth is less than the time-bandwidth required by equation (2.43). Subsequent further reduction of the lower passband edge in Fig 2.17(d) to 0.00625 gives even larger ripple with a slight deterioration at the edge of the transition band.

The reason for this algorithm exhibiting this behaviour has not been explored. It will, however, cause problems when attempting to generate non-linear phase filters.

2.5.2 Algazi-Suk

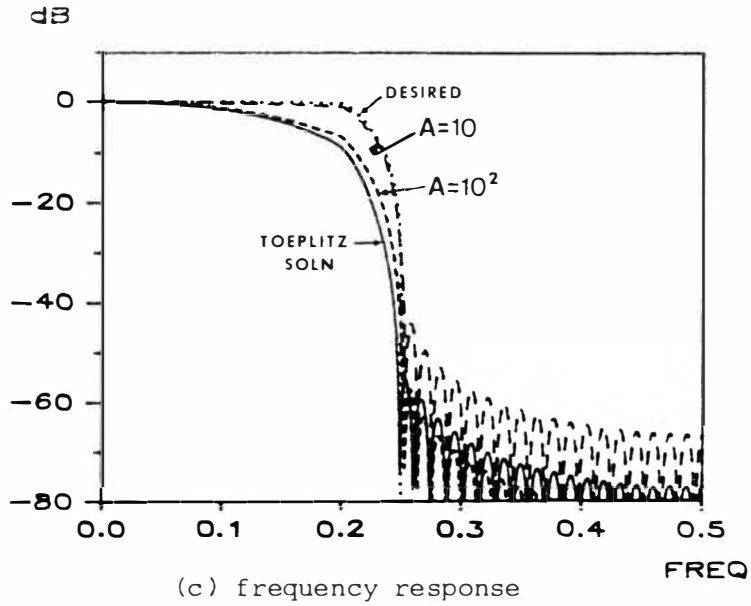
The results of Fig 2.9 are based on the solution of equation 2.34 using only 5 terms with C=8.0 in equation (2.44). The result of Fig 2.8 is based on an initial solution provided by the Toeplitz matrix inversion technique of method (v), followed by application of the iterative procedure of equation (2.35). This procedure requires the designer start with several attempts using method (v) with different weight values A_p, A_s in Fig 2.10(a). Subsequently, applying the iterative procedure, how many iterations are required and what value of A in equation (2.35) should be used?

Starting with the initial result of the lowpass example from method (v), Table VII and Fig 2.18 demonstrate the effect of 20 iterations of equation (2.35) with three values of A. A low value of A restores much of the passband region of the filter (Fig 2.18(a)) but at a cost of a significant decrease in the stopband attenuation. Increasing the value of A reduces the ability of the procedure to correct the premature fall of the stopband response and for A=10 an increase in stopband attenuation is achieved. From Fig 2.18(b) this increase is achieved after 2 iterations beyond which there is no further improvement.



(a) change in passband ripple

(b) change in stopband ripple



(c) frequency response

Fig. 2.18: Algazi's iterative algorithm with the Toeplitz solution as a starting estimate: the effect of different weight values A for the Lowpass example.

TABLE VII: APPLICATION OF 20 ITERATIONS OF THE ITERATIVE PROCEDURE OF ALGAZI TO THE RESULT FROM THE TOEPLITZ SOLUTION USING THE LOWPASS EXAMPLE WITH "A" AS THE PARAMETER.

	f_p	f_c	f_s	δ_p (dB)	δ_s (dB)
Desired	0.200	0.225	0.250		< -60
A = 10	0.0	0.222	0.255	*	-44.22
10^2	0.0	0.188	0.254	*	-57.42
10^4	0.0	0.172	0.253	*	-69.12

* see Fig. 2.18(a)

The discussion above raises several questions: why not apply the iterative procedure directly, using, as an initial starting estimate a truncated version of the model $h_i(t)$. The secondary questions are then what value A should be used and how many iterations are needed?

The first question has already been answered. Method (iv) of Table I reflects the result of the direct application of the iterative procedure. The result fails to meet the desired stopband specification. With regard to the choice of A, Algazi et al in their original work showed that as A ->∞, the procedure converges to $\phi_0(t)$, the zero order prolate spheroidal wavefunction. Thus a large value of A is inferred. Fig 2.19 demonstrates the convergence property of method (iv) with A as the parameter.

Comparing the result of Fig 2.18 with Fig 2.19, it appears that with a low value of A (A=10 in Fig 2.18), the solution is converging to that provided by the direct application of the iterative procedure in Fig 2.19 with $A=10^5$. The conclusion is that the application of the iterative procedure as a trimming procedure to an initial solution provided by the Toeplitz matrix inversion method is pointless because there is no real benefit over the direct application of the iterative procedure.

2.5.3 Cuthbert's Method

In Chapter 3, this algorithm will be discussed in more detail. However, when reporting his approach to the solution of a non-linear phase problem he raised the issue of weighting [Cuthbert, 1974b]

without indicating typical values of A_p and A_s that could be used This question is examined here briefly.

For the lowpass example of Fig 2.12(a), Table VIII shows the effect on the frequency parameters for different values of A_s. For the last case with $A_s=10^4$, the algorithm had difficulty in arriving at a solution when trying to solve equation (2.45). In addition, the final result produced a filter with loss because $|H(\omega)| = 0.71$ instead of 1.0.

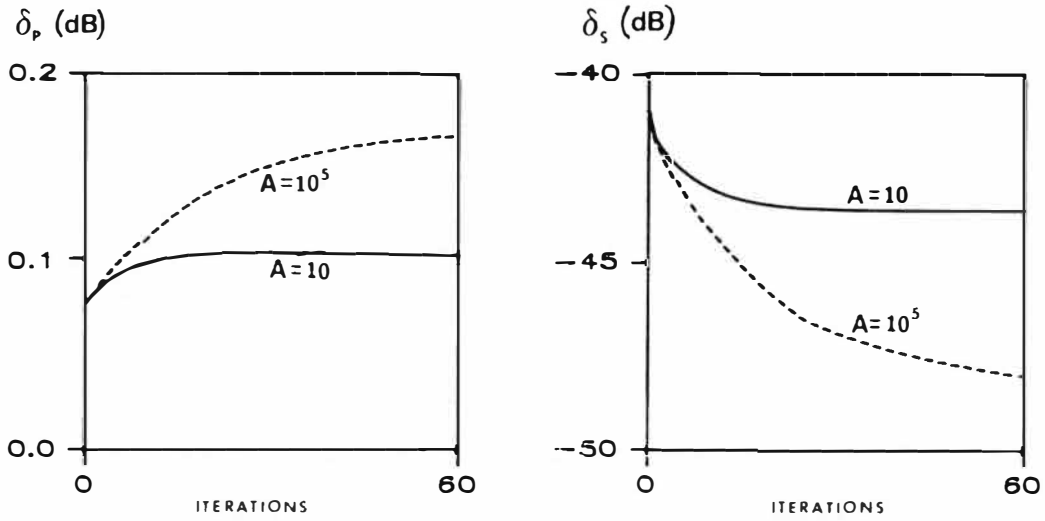
2.6 SUMMARY

This chapter has catalogued some of the more important approaches to solving the approximation problem for linear phase FIR filters and attempted to present their performance in a consistent and unified manner to bring out their relative strengths and weaknesses from a practical point of view. Of the five algorithms implemented, two stand out: the classical window technique and the McLellan-Parks algorithm.

It is clearly evident that the classical window methods offer very good results with the additional advantage that they are extremely simple to implement. Of the two windows demonstrated, the Kaiser window is to be preferred on two counts: it maintains the desired transition width more closely and is simpler to implement. Furthermore, many other well known windows such as the Hamming and Hanning windows that were examined but have not been discussed, all failed to meet the standard achieved by the Kaiser window.

The algorithm of McLellan-Parks offers the best result at a cost of a small amount of additional storage spaced and increased CPU time, but is mathematically complex to implement. The fact that the algorithm has been published and is widely available makes it potentially one of the most efficient design methods available for strictly linear phase filters. However, unlike the classical window technique, as indicated in section 2.5.1, there are hidden traps when the algorithm is used to generate Hilbert filters.

Cuthbert's method is reasonably successful but at a cost of very long solution times and very large array storage requirements. Both can be reduced by reducing the problem order. The order used for the example was 73x512 (Table IV) where 73 is the number of impulse



(a) passband (b) stopband

Fig. 2.19: Behaviour of Algazi's iterative algorithm for different weight values and number of iterations using the Lowpass example.

TABLE VIII: PERFORMANCE OF CUTHBERT'S METHOD AS A FUNCTION OF THE STOPBAND WEIGHT PARAMETER A_s.

A_p=1.0 , iter=3

	f_p	f_c	f_s	δ_p (dB)	δ_s (dB)	
Desired	0.200	0.225	0.250		< -60	
A = 10	0.200	0.226	0.254	0.158	-47.88	
10^2	0.198	0.225	0.253	0.247	-61.28	
10^3	0.190	0.222	0.252	0.248	-77.65	
10^4	0.208	0.218	0.252	-0.681	-92.16	**

** see text: final passband $|H(\omega)|$ = 0.71

response points and 512 is the number of points defining the frequency response. If a coarser grid on which the ideal frequency response is defined can be tolerated, then this will reduce both the required solution time and memory requirements.

The statistical approach where the filter is considered as a signal estimator and exploits certain mathematical symmetry properties of the problem to achieve a fast solution time in fact provides a result that is very poor when considered strictly as a filter. In situations where the best signal estimate of a noisy signal is required, such as in process control systems, this result may be quite adequate. In addition, the algorithm as implemented in this work was sensitive to the stopband weighting value. Both Farden and Algazi use $A_s = 1000$. This value gave very poor results when used here, while values of 10 and 100 gave a solution that was unusable. Stopband weights of 10^4 - 10^6 were found to be the most useful.

CHAPTER 3.0

NON-LINEAR PHASE FIR FILTERS

3.1 INTRODUCTION

This chapter expands the work of chapter 2.0 to consider the non-linear phase case. Repeating, for the sake of clarity, some equations of Chapter 2.0, recall the frequency response of a filter is described by

$$\begin{aligned} H(\omega) &= H_r(\omega) + jH_q(\omega) \\ &= |H(\omega)|e^{j\theta(\omega)} \end{aligned} \tag{3.1}$$

and that for the linear phase case, the constraint $H_q(\omega)=0$ was imposed. This constraint is now removed. Thus, introduction of a phase response complicates the approximation problem in two ways:

(a) two constraints - magnitude and phase must now be satisfied simultaneously.

(b) removing the constraint $H_q(\omega)=0$ now prohibits the impulse response sequence h(n) from exhibiting any symmetry. Furthermore, straight forward application of Fourier theory requires that for h(n) to remain a real sequence, the real and quadrature components of equation (3.1) must exhibit Hermitian symmetry:

$$\begin{aligned} H_r(\omega) &= H_r(-\omega) \\ H_q(\omega) &= -H_q(-\omega) \end{aligned} \qquad |\omega| \leq \pi/T \tag{3.2}$$

In Chapter 2.0, it was shown that the requirement of causality produced the linear phase term $\phi(\omega)=\omega(N-1)T/2$. This term is still present, so a non-linear phase filter has a total phase response

$$\phi_{tot}(\omega) = \omega(N-1)T/2 + \theta_D(\omega) \tag{3.3}$$

where $\theta_D(\omega)$ is the $\theta(\omega)$ of equation (3.1). Thus it is apparent that the filter phase response is a deviation $\theta_D(\omega)$ from the implicit linear phase base. All future discussion of phase response will refer only to $\theta_D(\omega)$ of equation (3.3).

3.2 EXISTING DESIGN METHODS

Cuthbert (1974b) recognised that given the symmetry requirements

of the frequency response in equation (3.2), the impulse response can be considered as the sum of two symmetric sets a(n) and b(n), whose DFT's correspond to the components of equation (3.1). For N even,

$$\begin{aligned} h(n) &= a(n) + b(n) \\ h(N-1-n) &= a(n) - b(n) \end{aligned} \qquad n = 0,\ldots,\frac{N}{2}-1 \qquad (3.4)$$

where N is the number of impulse response coefficients. Re-writing equation (2.4) for clarity:

$$H(\omega) = \sum_{n=0}^{N-1} h(n)e^{j\omega Tn} \qquad (3.5)$$

and substituting (3.4), it can be shown that the frequency response is given by

$$H(\omega) = e^{-j\omega(N-1)T/2}\left\{\sum_{n=0}^{\frac{N}{2}-1} 2a(n)\cos\{\omega T[n-[N-1]/2]\} - j\sum_{n=0}^{\frac{N}{2}-1} 2b(n)\sin\{\omega T[n-[N-1]/2]\}\right\} \qquad (3.6)$$

for N even. Similarly, separating the desired frequency response into real and quadrature components

$$D(\omega) = |D(\omega)|\cos\Phi(\omega) + j|D(\omega)|\sin\Phi(\omega) \qquad 3.7$$

then, using equations (3.6) and (3.7), Cuthbert solved this problem using the mean-square approach of equation (2.29) and a published optimisation algorithm [Fletcher et al, 1963] to separately solve for the sequences a(n) and b(n). In addition, he demonstrated that by performing a second optimisation run on each component over the stopband specification only, a useful improvement in stopband rejection was achieved at the expense of a very small increase in the passband ripple.

The equivalent formulation to equation (3.6) for N odd is similar, but requires an additional step in order to fit it into the same algorithm. Appendix 1 details the derivation of equation (3.6) for both odd and even N and shows the extra computational step for N odd.

Holt et al [1976] used the same approach as Cuthbert, but instead demonstrated the use of the Remez Exchange Algorithm which guaranteed the magnitude error will be within the prescribed tolerance. Neither method, however, gives precise control over the phase error.

Steiglitz [1981] instead approached the problem as a non-linear programming problem: find the h(n) in equation (3.5) so as to minimise the parameter λ in

$$\begin{aligned} &\left|\,|H(\omega_k)| - D_k\right| \leq \lambda TM_k \\ &|\mathrm{Arg}H(\omega_k) - \Phi_k| \leq \lambda TP_k \end{aligned} \qquad (3.8)$$

where D_k and Φ_k are the desired magnitude and phase and TM_k, TP_k are the set of magnitude and phase tolerances respectively. His solution was to convert the non-linear problem to a linear problem and use the well known simplex method of linear programming. This approach guaranteed both the magnitude and phase errors were within tolerance and that this approach was optimal to a first order approximation of the error.

3.3 IMPULSE RESPONSE LENGTH

For someone well versed in the practice of Fourier theory, it may be apparent that, in general, one penalty of requiring non-linear phase response is that a longer impulse response is required to maintain the same magnitude response of the equivalent linear phase filter. However, such an observation is not intuitively obvious.

The basic questions of "how much longer" and what is the relationship between the phase response and impulse response length remain largely unexplored. This section is an attempt to provide a partial answer to the questions raised. For clarity and simplicity, the discussion will be in terms of the continuous variables t (time) and ω (frequency).

Two cases will be considered:

(a) sine phase deviation

(b) square-wave phase deviation. This case is important here because the work on speech randomisation in Chapter 4.0 suggests this form of phase response is required. It is also the opposite extreme of the sine phase deviation.

The exception to the general rule is the "minimum phase" or minimum delay filters [Oppenheim,1975b]. These filters are more usefully described in terms of their z-transform and zero distribution. Herrmann [1970c] showed how to generate minimum phase FIR filters by selecting half of the zeroes of a prototype linear phase filter. However, while this procedure generates a non-linear phase response,

the designer has no control over it: the phase response is fixed by the positions of the remaining zeroes. Thus a minimum phase filter is a special case of the more general non-linear phase case. In this case only, the number of coefficients required for the impulse response is less than the equivalent linear phase filter to satisfy the same gain response. Later, Goldberg et al [1981] showed that this approach of zero selection gives the lowest possible order N and showed how the Remez exchange algorithm could be used to advantage. More recently, Kamp et al [1983] demonstrated a constrained approximation procedure for low-pass minimum phase filters that can be implemented via a simple modification to the McLellan-Parks algorithm, but that bandpass filters may in principle require more sophisticated means.

3.3.1 Sine phase deviation

A particular solution to this case is known [Papoulis,1962b], but is heavily disguised because it is presented in terms of the distortion products introduced into the output of a linear system. Here, the solution is generated and given in terms of the impulse response function of the system. Consider the (causal) specification (Fig. 3.1)

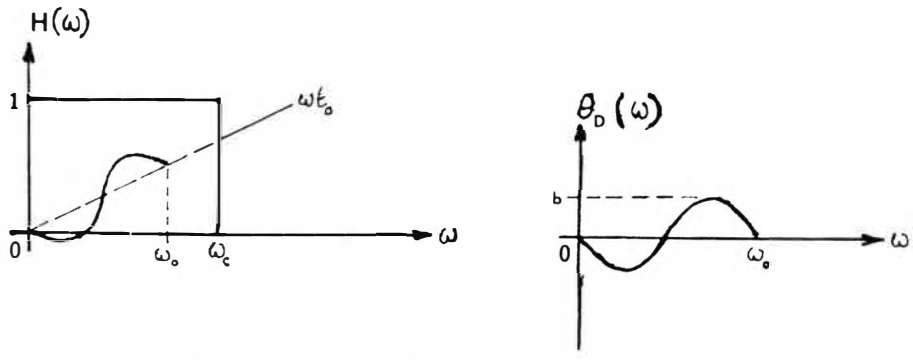


Fig. 3.1: Causal specification for the sine phase deviation.

$$A(\omega) = \begin{cases} 1.0, & 0 \leq \omega \leq \omega_c \\ 0, & \text{otherwise} \end{cases} \qquad (3.9a)$$

$$\Phi_T(\omega) = \begin{cases} \omega t_0 - b\sin\frac{2\pi\omega}{\omega_0}, & 0 \leq \omega \leq \omega_0 \\ \omega t_0, & \omega_0 < \omega \leq \omega_c \end{cases} \qquad (3.9b)$$

ie $H(\omega) = A(\omega)\exp\{-j\Phi_T(\omega)\}$ with $\theta_D(\omega) = b\sin(2\pi\omega/\omega_0)$

Then the impulse response h(t) is given by the inverse Fourier transform

$$h(t) = \frac{1}{\omega_c} \int_0^{\omega_c} H(\omega) e^{j\omega t} \, d\omega \qquad (3.10)$$

and the solution (Appendix 2) is

$$h(t) = \frac{1}{\omega_c - \omega_0} \left(\frac{\sin(\omega_c \tau)}{\tau} - \frac{\sin(\omega_0 \tau)}{\tau} \right) + \sum_{n=-\infty}^{\infty} J_n(b) \frac{\sin\left(\tau + [2\pi n/\omega_0]\right)\omega_0}{\left(\tau + [2\pi n/\omega_0]\right)\omega_0} \qquad (3.11)$$

where $\tau = (t - t_0)$ and the $J_n(b)$ are Bessel functions of the first kind.

The particular solution usually described is for $\omega_0 = \omega_c$. In this case, equation (3.11) collapses to

$$\sum_{n=-\infty}^{\infty} J_n(b) \frac{\sin\left(\tau + [2\pi n/\omega_c]\right)\omega_c}{\left(\tau + [2\pi n/\omega_c]\right)\omega_c} \qquad (3.12)$$

Equation (3.11) says:

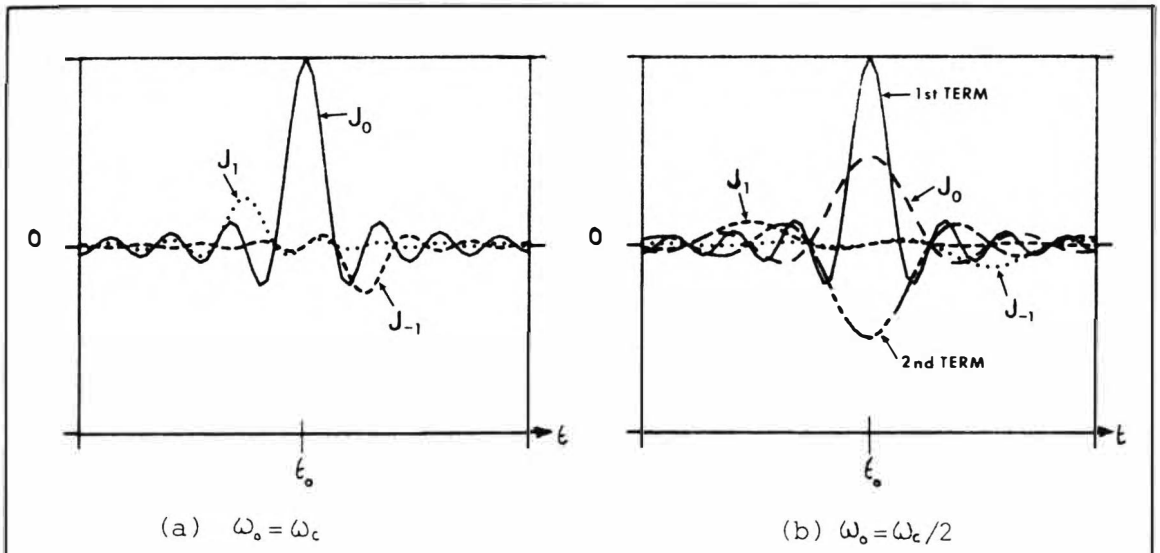
(i) in theory, the impulse response is infinitely long with sinc functions of peak amplitude $J_n(b)$ spaced evenly every $2\pi n/\omega_0$ in time, with the spacing and width controlled by ω_0.

(ii) in practice, the number of important terms is governed by the magnitude of b. If b is small (<<1), then only the first 3 Bessel terms J_{-1}, J_0, J_1 are significant.

(iii) when $\omega_0 < \omega_c$, the first two terms in equation (3.11) appear and the second term in ω_0 has a wider mainlobe than the first.

Fig. 3.2 shows each term in equation (3.11) for b=0.5 and the first 3 Bessel terms J_{-1}, J_0, J_1. If, as a first approximation, the mainlobe widths between the first zero crossings are used as the basis of the required impulse response length, a linear phase filter only contains the term $\text{sinc}(\omega_c t)$, so the mainlobe width is $t_{lp} = 2\pi/\omega_c$. In contrast, in equation (3.11) where $\omega_0 < \omega_c$ the extent of the impulse respone is dominated by the sinc function associated with the $J_n(b)$ terms as shown in Fig. 3.2(b). Using the outer end points of the mainlobes of the J_{-1} and J_1 terms, the required length is

$$t_{nlp} = 2 \times \frac{\pi}{\omega_0}(2n+1) \qquad (3.13)$$



(a) $\omega_o = \omega_c$ (b) $\omega_o = \omega_c/2$

Fig. 3.2: Sine phase deviation - components of the impulse response: b=0.5, f_c=0.125 (normalised frequency).

For the specific case of $\omega_o=\omega_c$ (equation (3.12)), then $t_{nlp} = 6\pi/\omega_c$ as evident from Fig. 3.2(a). This is 3 times the length required for the linear phase case.

3.3.2 Square-wave phase deviation

Consider the (causal) specification (Fig. 3.3)

$$A(\omega) = \begin{cases} 1.0, & 0 \leq \omega \leq \omega_c \\ 0 \ , & \text{otherwise} \end{cases} \qquad (3.14a)$$

$$\Phi_T(\omega) = \begin{cases} \omega t_o + b \ , & 0 \leq \omega \leq \omega_c/2 \\ \omega t_o - b \ , & \omega_c/2 < \omega \leq \omega_c \\ \omega t_o \ , & \text{otherwise} \end{cases} \qquad (3.14b)$$

ie $H(\omega) = A(\omega)\exp\{-j\Phi_T(\omega)\}$ and $\theta_D(\omega) = \pm b$

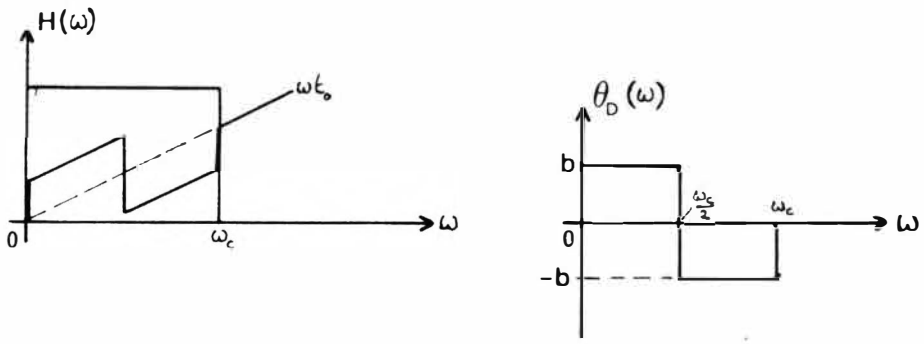


Fig. 3.3: Specification for causal square-wave phase deviation.

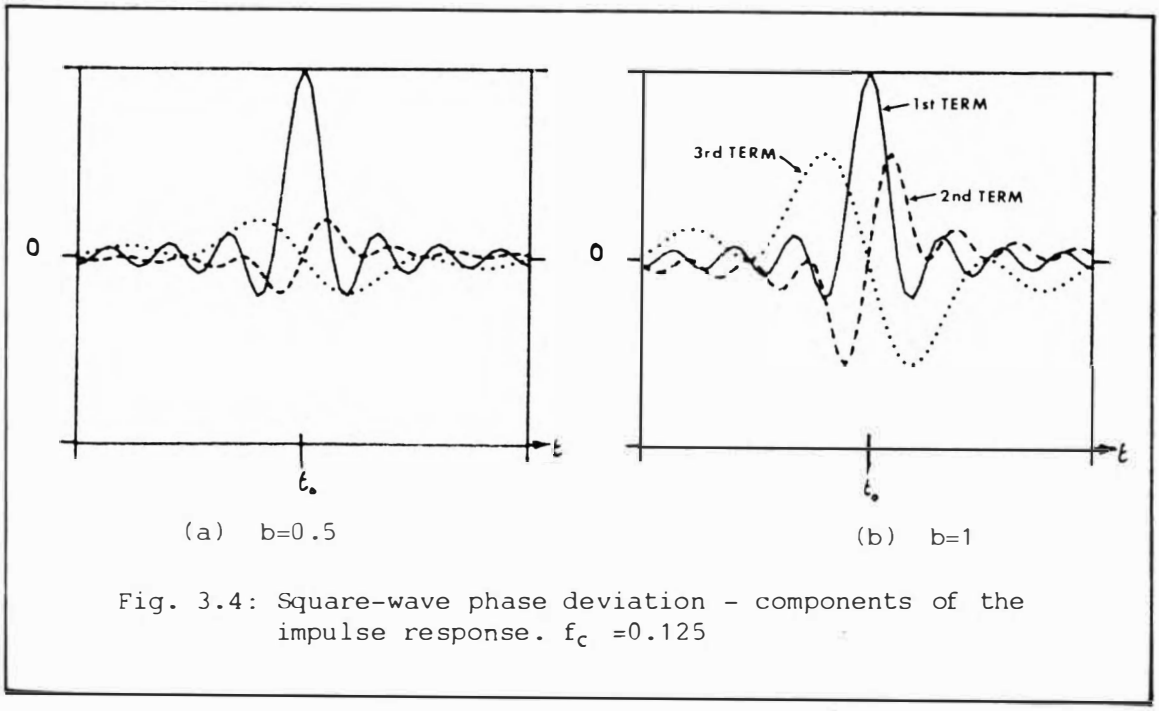
In the same manner as the previous section, the impulse response is given by (Appendix 3)

$$h_{sq1}(t) = 2\cos b\frac{\sin(\omega_c\tau)}{\omega_c\tau} + \sin b\left[\frac{\sin^2(\omega_c\tau/2)}{\omega_c\tau/2} - \frac{\sin^2(\omega_c\tau/4)}{\omega_c\tau/4}\right] \qquad (3.15)$$

$$\tau = (t-t_o)$$

Equation (3.15) is deliberately presented in this manner to emphasise the sinc-function components. Compared with the previous case, this form of phase deviation only requires three terms to completely describe the impulse response. For $b=\pm n\pi$, then h(t) collapses to the first term, which for n=0 represents the normal linear phase filter. Fig. 3.4 depicts the three components of equation (3.15) for b=1. Note the last term in $\omega_c/4$ has a mainlobe width four times greater than the first term. Therefore, using the mainlobe width as a first approximation to the required impulse response length, this square-wave deviation requires a length at least four times greater than the equivalent linear phase filter - in contrast to the sine deviation which only required a 3 times increase.

Also notice in Fig. 3.4b, increasing the phase deviation b causes an increase in the amplitude of the second and third components of equation (3.15). For b=1 as shown, the secondary lobes beyond the main lobes have a significant amplitude and are therefore likely to contribute significantly to the overall frequency response. Thus, the



(a) b=0.5 (b) b=1

Fig. 3.4: Square-wave phase deviation - components of the impulse response. f_c =0.125

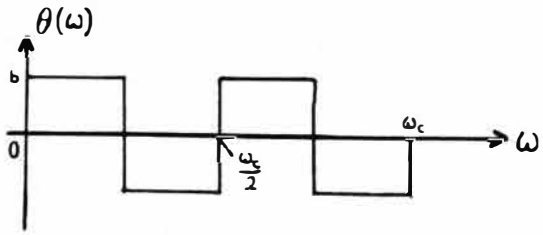


Fig. 3.5: Two square-wave phase cycles in the passband.

original extimate of a 4x increase is conservative.

Extending the above argument one step, consider the expanded phase specification of Fig. 3.5 so that two square-wave "phase cylces" are included:

$$\Phi_T(\omega) = \begin{cases} \omega t_o + b, & 0 \leq \omega \leq \omega_c/4 \\ \omega t_o - b, & \omega_c/4 < \omega \leq \omega_c/2 \\ \omega t_o + b, & \omega_c/2 < \omega \leq 3\omega_c/4 \\ \omega t_o - b, & 3\omega_c/4 < \omega \leq \omega_c \end{cases} \qquad (3.16)$$

Following the identical procedure in Appendix 3, it can be shown that

$$h_{sq2}(t) = 4\cos b\frac{\sin(\tau\omega_c)}{\tau\omega_c} + \frac{8\sin b}{\tau\omega_c}\Big(\sin^2(\tau\omega_c/2) - 2\sin^2(\tau 3\omega_c/8) + 2\sin^2(\tau\omega_c/4) - 2\sin^2(\tau\omega_c/8)\Big) \qquad (3.17)$$

Note this time that last term involves $\omega_c/8$ - ie a mainlobe width eight times longer than the basic linear phase term.

Comparing the sine and square-wave deviations, the following general comparisons and conclusions can be made:

(i) In theory, the sine phase deviation requires an infinite number of sinc components, while the square-wave deviation has a finite number (2*M+1) of components where M is the integral number of whole "phase cycles" included.

(ii) The increase in impulse response length required is determined in a complex manner by the peak phase deviation b, the number and extent of the "phase cycles" and the form of the phase response.

(iii) practically, for $\omega_o = \omega_c$ in the sine deviation case and for the specification of equation (3.14) for the square-wave deviation, the required increase in impulse response length for the sine deviation is small, but for the square wave phase

deviation it is much greater (demonstrated below).

(iv) Given that a square wave is composed of an infinite number of (odd order) sine waves, the two cases examined can be viewed as two opposite extremes and so any general phase deviation will require an impulse response length somewhere between the two extremes.

3.3.3 Windowing and Non-linear Phase

The discussion in section 2.3.2 on window methods showed the effect of applying a window to constrain the (time) length of the filter. Applying the same argument here by using equations (3.1) and (2.22) in equation (2.22b), then

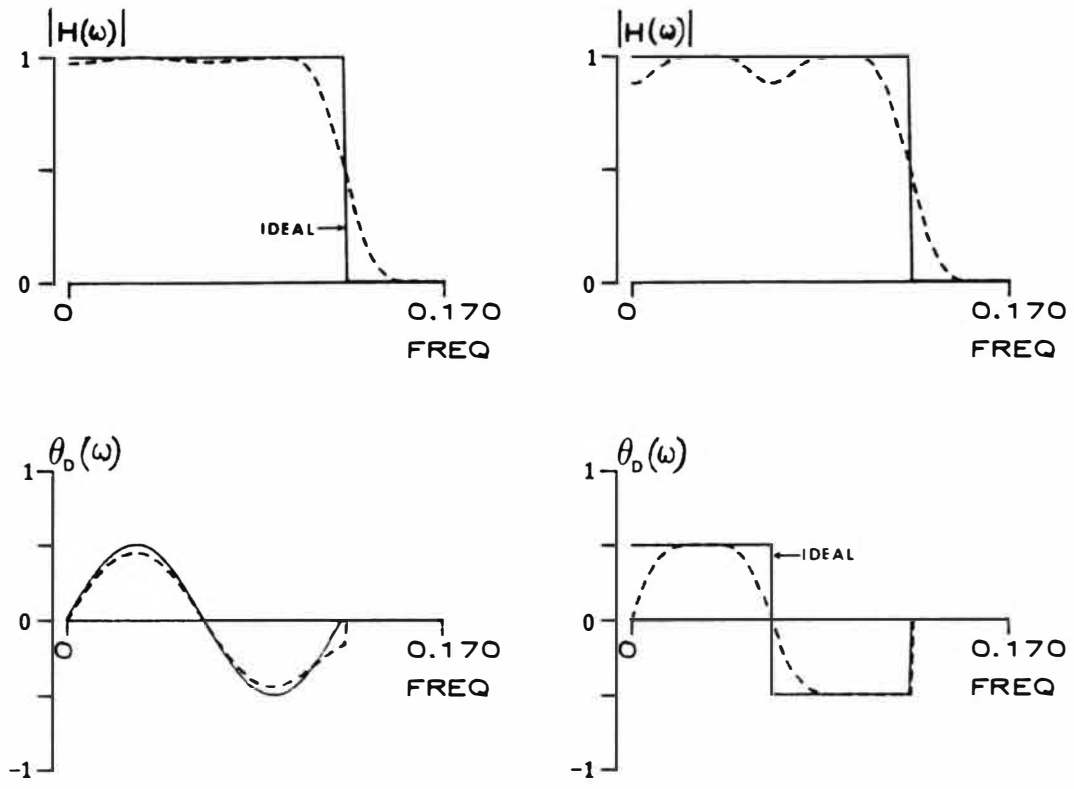
$$H_w(\omega) = H_r(\omega) * \frac{2\sin\omega T}{\omega} + jH_q(\omega) * \frac{2\sin\omega T}{\omega} \qquad (3.18)$$

Both components of the total frequency response undergo "smearing". Recalling that such smearing is most noticeable in the vicinity of a discontinuity, it is obvious that if either $H_r(\omega)$ or $H_q(\omega)$ in equation (3.18) above contain discontinuities to satisfy a phase requirement, there will be considerable degradation of both the amplitude and phase performance of the filter. From this observation, it can be inferred that the square-wave phase deviation of Fig. 3.3 will suffer more distortion than the sine-phase deviation of Fig 3.1 . For example, using a "brick-wall" specification with (normalised) frequencies f_c =0.125 and the phase specification of Fig. 3.3 with $f_o=f_c$, b=0.5, Fig 3.6 demonstrates the effect of the Kaiser window to restrict the impulse response to N=73.

As expected, the sine phase deviation filter shows small errors in the magnitude and phase, but the square- wave phase exhibits considerable error in the achieved magnitude and phase. This supports observation (iii) of section 3.3.2 above. It is apparent therefore, that the window technique is not, as a general rule, useful as a design technique for non-linear phase filters.

3.4 SOME EXISTING DESIGN METHODS IMPLEMENTED

Based on the discussion so far, there are three potential design algorithms that have been used in the literature for desinging non-linear phase filters: the Remez exchange algorithm; Cuthbert's



(a) sine phase deviation (b) square-wave phase deviation

Fig. 3.6: Effects of windowing in the presence of a non-linear phase using Kaiser's window with f =0.125, b=0.5.

numerical optimisation technique and Steiglitz's linear programming method. The Remez algorithm is, in theory, available through Mclellan's program and Cuthbert's method has been programmed for the examples in Chapter 2, so they are immediately available. Steiglitz's method requires considerable mathematical manipulation and, as he said, a special formulation of the Simplex algorithm. The simplex algorithm is also a multi-dimensional numerical optimisation technique. Thus, it is likely that Steiglitz's method will require a similar order of magnitude in CPU time for a solution as Cuthbert's method. Thus, an implementation of Steiglitz's method is not attempted. Instead, the available, already programmed algorithms are investigated.

3.4.1 McLellan-Parks Algorithm

Equation 3.4 showed that the composite impulse response can be formed by the sum of two symmetric sets of sequences. Thus, in principle, if the real and quadrature parts of the desired frequency

response are mapped consecutively onto the dense grid (array GRID) of the Mclellan-Parks algorithm, it should be capable of generating non-linear phase filters.

However, bearing in mind section 2.5.1 where it was pointed out that in some cases the algorithm has difficulty with Hilbert filters, all attempts to get this algorithm to provide useful results failed. Most of the failures occurred when trying to solve the quadrature part of the frequency response (ie requesting an odd symmetric sequence). After a short attempt to try and establish the reasons for the algorithm's failure, it was abandoned from any further consideration.

3.4.2 Cuthbert's Method

The Fortran subroutine as set out in Appendix 1 was developed from the outset to be able to handle both linear and non-linear phase filters. The results of Chapter 2 demonstrated the success of the method for linear phase filters at a cost of significant computer time and storage requirements. For a non-linear phass filter, the required computer time for a solution will double because the routine must perform two optimisation runs.

Using the common example of Fig 2.12(a) in Chapter 2, Table 3.1 shows the results for the sine phase deviation of Fig. 3.1 with $\omega_o=\omega_c$. Table 3.2, Fig. 3.7 and Fig. 3.8 show the result for the square-wave phase deviation of Fig. 3.3 as a function of N and b. In both cases, the same parameters used to give the result for Cuthbert's method in Table 2.1 are used here as well.

TABLE 3.1 CUTHBERT'S METHOD APPLIED TO THE COMMON LOWPASS EXAMPLE WITH SINE PHASE DEVIATION

parameters: $A_p=1$, $A_s=100$, iter=3, NFFT=1024

	desired	N = 73 b=0.5	N = 73 b=1
f_p	0.200	0.198	0.198
f_c	0.225	0.225	0.225
f_s	0.250	0.253	0.253
δ_p (dB)		0.227	0.183
δ_s (dB)	≤-60	-61.30	-61.37
CPU TIME		2:10.83	1:55.34

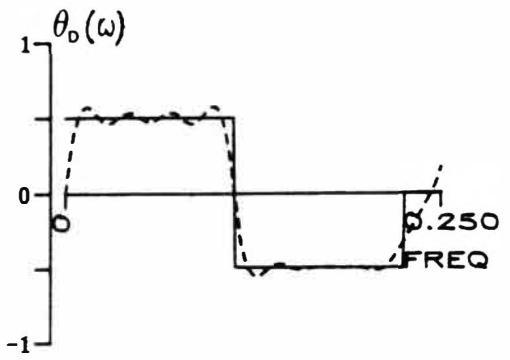
The results for Table 3.1 use only N=73, the impulse response length required for the linear phase case, in spite of the argument of section 3.3.1 which suggested that a 3x increase would be necessary. Comparing these performance figures with those of the entry for Cuthbert in Table 2.1 show no degradation in performance as a result of introducing the non-linear phase response. The reason for this is simply that on the basis of the arguments in section 3.3.1, the length N=73 is already longer than predicted would be required: for this example, the (normalised) mainlobe width of a linear phase filter is 4.4 ($1/f_cT$ = 4.44 for T=1). It was suggested that 3x4.44=13.32 would be needed. The normalised time length of the filter is 73 which is 5.48 times longer than predicted. The result for CPU time above would indicate that the problem was easier to solve for b=1 than for b=0.5. In fact, plots of the resultant phase were indistinguishable from the desired specification.

TABLE 3.2 CUTHBERT'S METHOD APPLIED TO THE COMMON LOWPASS EXAMPLE WITH SQUARE WAVE PHASE DEVIATION.

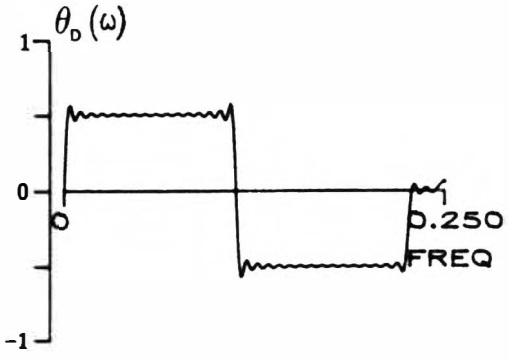
parameters: A_p =1, A_s =100, iter=3, NFFT=1024

	desired	b = 0.5		b = 1
		N=73	N=291	N=291
f_p	0.200	0.208	0.207	0.224
f_c	0.225	0.225	0.225	0.225
f_s	0.250	0.253	0.254	0.254
δ_p (dB)		1.1	1.01	3.29
δ_s (dB)	≤-60	-60.95	-77.39	-75.98
CPU TIME		1:57.27	30:10.46	30:23.41

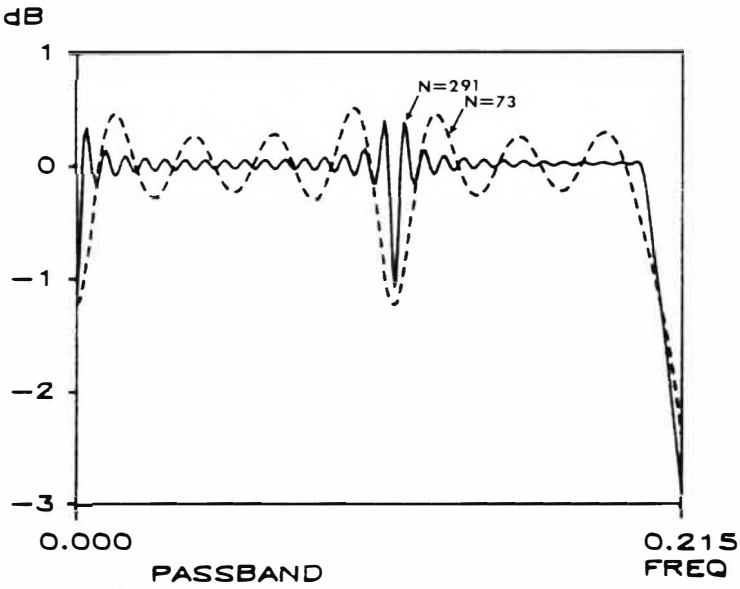
The approximation in section 3.3.2 requiring a 4x increase in N is reasonably accurate for both the phase and amplitude response as shown in Fig. 3.7. However, in Fig. 3.7(c) it is noteworthy that the large dip in the passband response occurs at the frequency of the sharp phase transition, and its depth is largely unaffected by the increase in N. That this should be so is intuitively correct if the time-bandwidth constraint of equation (2.10) is recalled. Furthermore, the sharp phase transitions correspond to discontinuities in the real and



(a) phase response compared with spec for N=73.



(b) phase response with N=291



(c) passband response achieved for N=73 and N=291

Fig. 3.7: Cuthbert's method applied to the square-wave phase deviation: resulting phase response for different filter lengths N and b=0.5 using the common lowpass example of Fig. 2.12(a) and the phase specification of Fig. 3.3.

quadrature components of the frequency response. It is well known that any truncation of a sequence whose Fourier components contain discontinuities introduces ringing (Gibbs phenomenum). Hence the ringing evident in the phase responses and the passband in the regions of the phase transitions.

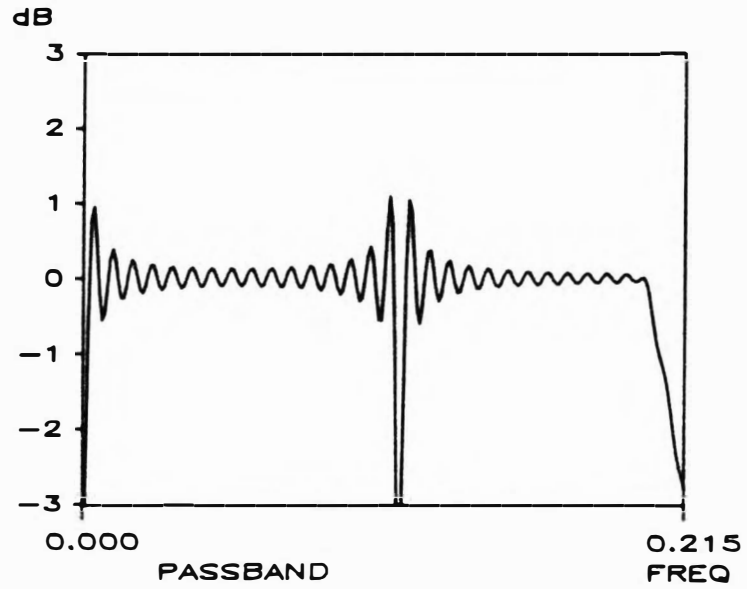


Fig. 3.8: Cuthbert's method for the same example in Fig. 3.7 but this time with b=1.0: resulting passband for N=291.

Fig. 3.8 demonstrates the effect of doubling the phase deviation from 0.5 to 1.0 for N=291. The resultant phase response is unchanged, but there is now a very large dip in the passband at the phase transition frequency which requires excessively large values of N to begin to correct.

Finally, for N=291, the order of the problem is 291x512. Notice the very large increase in CPU time required for a solution.

3.5 ALGAZI-SUK'S ITERATIVE METHOD REVISITED

The original work that developed the iterative procedure of equation (2.35) was restricted to linear phase filters. However, with appropriate choice of the FFT routine, the procedure is directly applicable to the non-linear phase case without any modification.

It has been established that a non-linear phase repsonse precludes any symmetry in the impulse response of the filter. Thus the FFT routine used must explicitly retain both halves of the impulse response. The algorithm FFT842 has the required characteritics. In performing the bandlimiting operation, it is important to ensure that the real and quadrature parts of the frequency response are limited identically.

Using the same example from the previous section, Algazi's method gives almost identical results as those for Cuthbert's method for f_p, f_c, f_s and δ_p. The phase and passband responses of Fig. 3.7 and 3.8 are duplicated. The real difference between the two methods is in the resulting stopband ripple and the CPU time required for a solution, as shown in Table 3.3.

TABLE 3.3 STOPBAND RIPPLE AND CPU TIME FOR ALGAZI'S METHOD COMPARED WITH CUTHBERT'S USING THE SQUARE-WAVE PHASE.

parameters: NFFT=1024, $A=10^5$, iterations=20

	δ_s (dB)	
	ALGAZI	CUTHBERT
b=0.5, N=73	-43.61	-60.95
b=0.5, N=291	-61.66	-77.39
b=1.0, N=291	-61.25	-75.98
CPU TIME	12.6 s	30:23.41 (N=291)

The sine phase deviation is not shown because it has been demonstrated that Alazi's method gives identical results to Cuthbert's method for the phase and passband responses.

3.6 RANDOM PHASE FILTERS

3.6.1 Background

Here, the requirements of Chapter 5.0 are anticipated, but since the emphasis of the discussion will be on filter design and will rely on the work of earlier sections, it is included here.

Chapter 5.0 will show that a speech band lowpass filter with ω_c= 3400 Hz and a random phase response such as that shown in Fig. 3.9 is desirable.

The phase response is constrained to lie between ω_x and ω_y such that $0 < \omega_x < \omega_y < \omega_c$ assuming a lowpass filter. Each phase band has a frequency width $\Delta\omega_\theta$ and is centred on harmonic multiples $\ell\omega_\theta$, $\ell=1,..L$. For the work of Chapter 5.0, the most important criteria of the filter performance is the passband ripple δ_p. In order to avoid modification of the amplitude spectrum of the speech signal, δ_p should

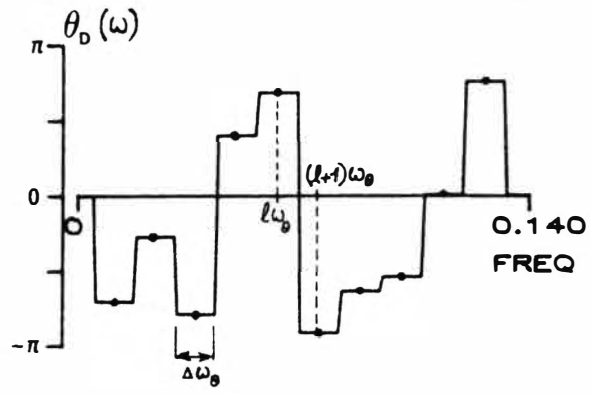


Fig. 3.9: "Ideal" desired phase response

be as small as possible. An arbitrary design maximum of $\delta_p = \pm 2$ dB is set here. In the discussion that follows, it will be shown that given a random number sequence $\{r_n\}$, there are two ways that this sequence can be applied to generate the phase response and that the choice of method has a noticeable effect on the resulting passband and stopband ripple. It will also be shown that the ideal response of Fig. 3.9 can be considered as an extension of the square-wave phase response of section 3.3.2. It will be shown that the phase band-width $\Delta\omega_\theta$ is the dominant influence on the required number of the impulse response coefficients and that a prohibitively large impulse response length is required. In practice, phase accuracy will have to be sacrificed in order to get practically useful filter lengths and to meet the desired passband ripple specification.

It has been demonstrated that the only difference in performance between Algazi's and Cuthbert's methods is that Cuthbert's method in general gives better (lower) stopband ripple. Thus, in view of the very long solution times required by Cuthbert's method, Algazi's method is chosen to the coefficients of the random phase filters.

3.6.2 Basic Filter Specification for Discussion

Using the common definition of frequency response parameters set out in section 2.2.2, the normalised specification of the filter that will form the basis for discussion is shown in Fig. 3.10

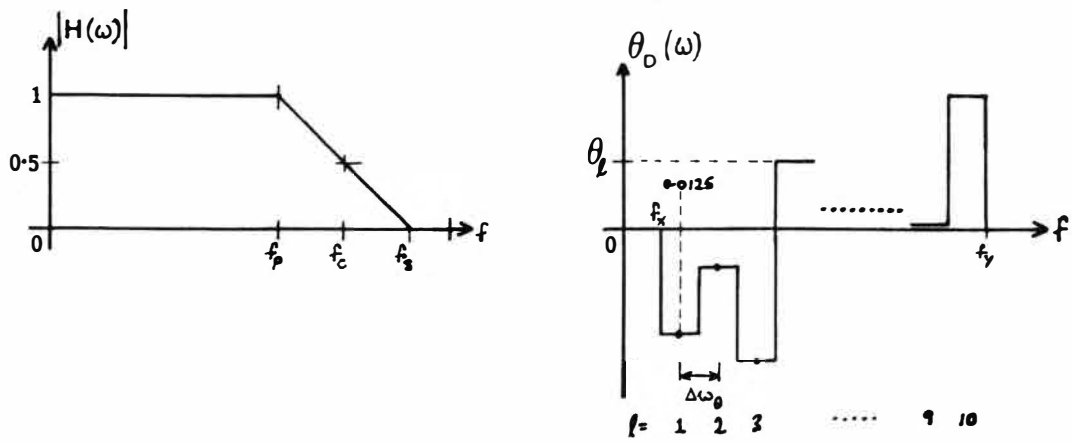


Fig. 3.10: Example used for discussion: amplitude and phase specification. θ_ℓ is given by equation (3.20).

$f_1 = 0.4$ $\quad f_x = 0.00625$

$f_c = 0.425$ $\quad f_y = 0.13125$

$f_2 = 0.45$ $\quad \Delta\omega_\theta = 0.0125$

$\delta_p \leq 2$ dB $\quad \ell\omega_\theta = 0.0125\ell,\ \ell = 1,\ldots,10$

$\delta_s \leq -40$ dB

For a sampling rate of 8 KHz, these frequencies translate to f_c = 3400 Hz with a transition width of 400 Hz. The frequency width of each phase band is 100 Hz and is centred on multiples of 100 Hz. The manner in which the phase magnitudes $\theta_\ell(\omega)$ are established is described in the section that follows.

The (normalised) transition width of 0.05 of the common example of Chapter 2.0 has been retained, so the arguments of section 2.4.3 that suggested N=73 for a linear phase filter apply here. However, based on the foregoing discussion so far, it is obvious that for the same N=73, there is no possibility of achieving the same stopband performance as the linear phase examples. Thus, the original stopband specification is relaxed here and arbitrarily set to $\delta \leq -40$ dB.

3.6.3 Random Phase Generation

The magnitudes $\theta_\ell(\omega)$ of each phase band are derived from a uniform distribution random number generator that generates a sequence $\{r_n\} \in [0,1)$. These r_n are modified to give

$$r'_n = (2r_n - 1)\pi \qquad (3.19)$$

so that $\{r'_n\} \in (-\pi, \pi)$. There are two ways the sequence r'_n can be assigned to $\theta_\ell(\omega)$:

(a) $\theta_\ell(\omega) = r'_\ell$ (3.20a)

(b) $\theta_\ell(\omega) = \theta_{\ell-1}(\omega) + r'_\ell$ (3.20b)

$\ell = 1, \ldots, L$

However, because of the manipulation performed in equation (3.19), it possible to get

$$|r'_n - r'_{n-1}| > \pi \tag{3.21}$$

Thus, for option (a), it is possible to have

$$|\theta_\ell(\omega) - \theta_{\ell-1}(\omega)| > \pi \tag{3.22a}$$

but for option (b)

$$\theta_\ell(\omega) - \theta_{\ell-1}(\omega) = r'_n \tag{3.22b}$$

where $|r'_n| \leq \pi$ from equation (3.19).

In addition, option (a) has a phase response that is restricted to the principal interval $-\pi < \theta_\ell(\omega) < \pi$, but option (b) can range over $-p\pi < \theta_\ell(\omega) < p\pi$, $p > 1$. Later on, it will be demonstrated that using option (b) achieves a lower final passband and stopband ripple than version (a). That option (b) should give a better behaviour of the filter amplitude response is assumed to be because the condition of equation (3.22a) never arises. However, this aspect has not been pursued; rather the phenomenon has simply been observed and is reported here.

For this discussion, option (a) (equation 3.20a) is used to assign the random phase values at the $\theta_\ell(\omega)$ points.

3.6.4 Establishing the impulse response length

Given the phase specification of Fig. 3.10(b), it is desirable to be able to establish a rough estimate for the required impulse response length, but it is not immediately obvious that the earlier discussion of section 3.3.2 is of any help.

However, in Appendix 3, when deriving the solution to equation (3.15) describing the square-wave phase response, the derivation considered the frequency response of the filter as the sum of two components. Expanding on this approach, the frequency response of Fig.

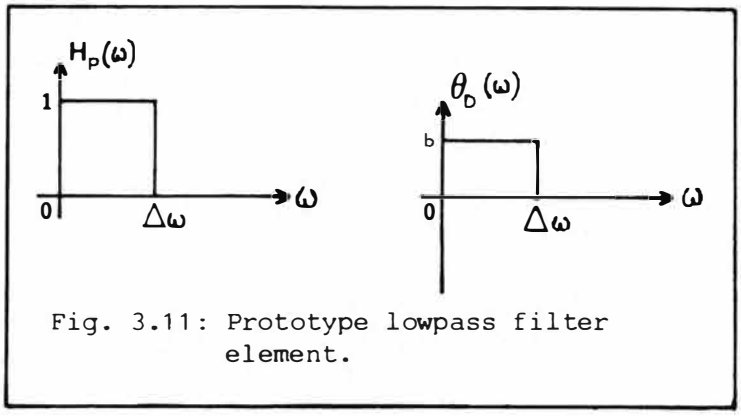


Fig. 3.11: Prototype lowpass filter element.

3.10 can be considered as the sum of appropriately frequency translated causal prototype lowpass filters of the form of Fig. 3.11. The prototype filter has a constant phase response b over its interval $\Delta\omega_\theta$ and $|H_p(\omega)| = 1.0$. The impulse response $h_p(t)$ of this prototype filter is (Appendix 4):

$$h_p(t) = \cos(\Delta\omega_\theta \tau/2 + b)\frac{\sin(\Delta\omega_\theta \tau/2)}{\Delta\omega_\theta \tau/2} \qquad (3.23)$$

where $\tau = t - t_o$

Note the sinx/x term is in $\Delta\omega_\theta/2$. For the phase response of Fig. 3.3, $\Delta\omega_\theta = \omega_c/2$, and thus the sinx/x term in equation (3.23) is in terms of $\omega_c/4$ which matches the last term of equation (3.15). Hence, from equation (3.23) and using the main-lobe width argument of section 3.3.2, it is possible to establish an estimate for the required impulse response given by

$$N_{nlp} = N_{lp} * 2\omega_c/\Delta\omega_\theta \qquad (3.24)$$

where N_{lp} is the linear phase estimate established in the manner of section 2.4.3.

However, equation (3.24) is not realistic because N_{nlp} is apparently dependent on ω_c. Equation (3.24) suggests that two filters with identical phase response but one with half the (amplitude) bandwidth of the other requires only half as many points in its impulse response. Such a suggestion is wrong, because N_{lp} is dependent on the transition bandwidth of the filter: two linear phase filters with identical transition bandwidth but where $\omega_{c2} = \omega_{c1}/2$ have the same N_{lp} to give the same δ_p and δ_s result. The problem is compounded because equation (3.24) takes no account of the phase deviation b and earlier sections showed that this parameter has a significant effect

on the resulting amplitude response.

As discussed in section 3.3.2, the root of the problem lies in the time-bandwidth constraint of equation 2.10. An heuristic approach base on the sinx/x term of equation (3.23) that has been found useful is to use

$$\tilde{N}_{nlp} \approx N_{lp} x 2(f_s - f_p)/\Delta f_\theta \qquad (3.25)$$

where the f's in equation (3.25) are the normalised parameters of Fig.3.10. Equation (3.25) suggests $\tilde{N}_{nlp} \approx 585$ which, on the basis of the discussion in section 3.3.2 is likely to give a reasonable approximation to the desired phase response, but the passband will contain significant ripples. Fig 3.12 and Table 3.4 indicate the result achieved.

TABLE 3.4: RESULT OF APPLYING EQUATION 3.25 THE SPECIFICATION OF FIG. 3.10 USING ALGAZI'S METHOD.

parameters: NFFT=1024, A=10^5, iterations=20

	desired	result
f_p	0.400	0.420
f_c	0.425	0.425
f_s	0.450	0.450
δ_p (dB)	≤ 2.0	2.79
δ_s (dB)	≤-40.0	-63.79

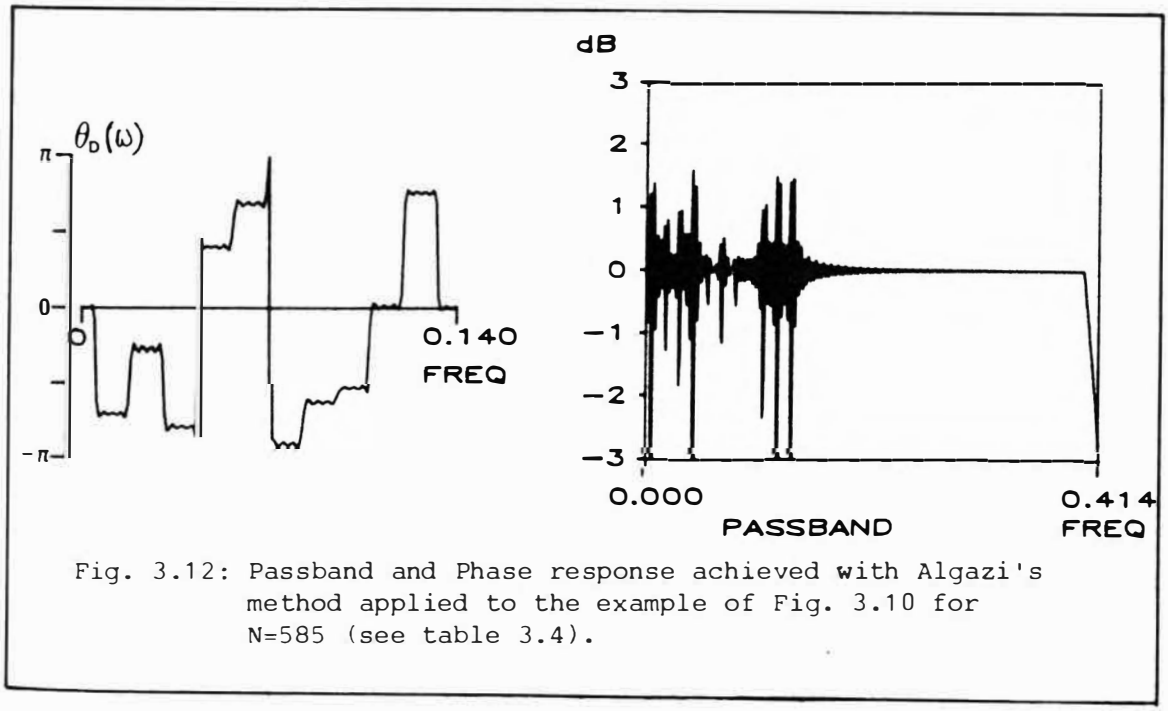


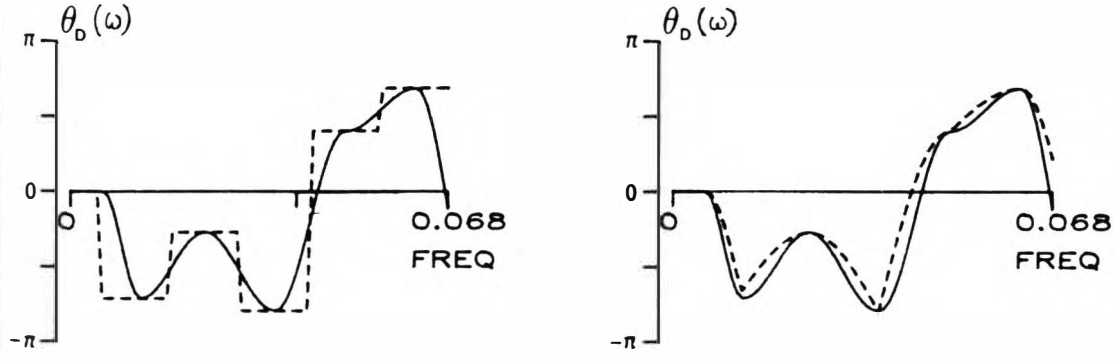
Fig. 3.12: Passband and Phase response achieved with Algazi's method applied to the example of Fig. 3.10 for N=585 (see table 3.4).

Comparing the passband responses of Fig 3.11 with 10 "phase bands" and Fig. 3.7 with 2 "phase bands", it is evident that $\Delta\omega_\theta$ controls the degradation of the passband via the number of phase transitions that occur. Decreasing the width of $\Delta\omega_\theta$ is only going to compound the problem. Therefore, for practically useful filters, especially where $\Delta\omega_\theta$ becomes small, the ideal phase response of Fig. 3.10 must be sacrificed.

3.6.5 Sacrificing Phase Accuracy

The question addressed here is what modification to the ideal phase response of Fig. 3.10 should be made to allow the passband ripple specification to be met without having to use very long impulse response sequences. Essentially, the sharp phase transitions must be replaced with a smooth phase transition, that by implication, has some frequency width.

Based on the examination of the sine phase deviation of section 3.3.1, two approaches were considered. Using the defined frequency $\ell\omega_\theta$ points of Fig 3.9, fit either a sin() or $\sin^2()$ function between these points. Retaining for the use of option (a) of equation 3.20a to assign the phase values, Fig 3.13(a) is an expansion of the first 5 phase bands of Fig. 3.10 with the $\sin^2()$ function used. Fig 3.13(b) shows the difference between the sin() and $\sin^2()$ fit and shows that the $\sin^2()$ fit preserves more of the original phase "area". In addition, it is smoother at the vertices (the $\ell\omega_\theta$ points) in contrast to the sin() fit which shows a discontinuity in changing direction at some of the vertices.



(a) $\sin^2()$ fit compared to the ideal.

(b) difference between sin() fit and $\sin^2()$ fit.

Fig. 3.13: Alternative ways of approximating the ideal desired phase specification of Fig. 3.9.

Experiment has shown that the $\sin^2()$ fit gives the better passband and stopband ripple results. Thus the $\sin^2()$ is chosen. In particular, the constraint that the phase response lie in the range $\omega_x - \omega_y$ is retained, and for

$\theta_{\ell+1} < \theta_\ell$:

$$\theta(\omega) = \theta_\ell(\omega) - R\sin^2(\pi\omega/2\Delta\omega_\theta) \qquad (3.26a)$$

$$\omega = \ell\omega_\theta, \ldots, (\ell+1)\omega_\theta$$

$$\ell = 1, \ldots, L$$

$\theta_{\ell+1} > \theta_\ell$:

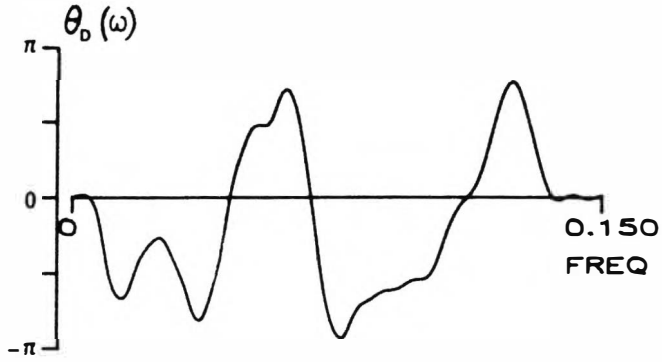
$$\theta(\omega) = \theta_\ell(\omega) + R\sin^2(\pi\omega/2\Delta\omega_\theta) \qquad (3.26b)$$

where $R = |\theta_{\ell+1} - \theta_\ell|$.

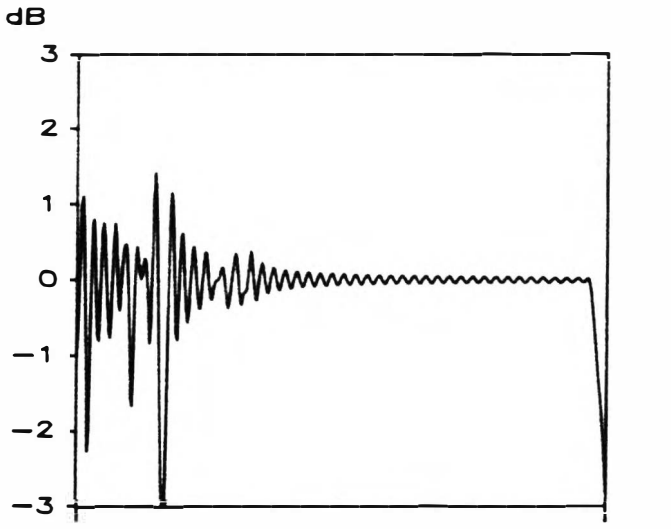
For the first and last half intervals, the curve is fitted between $\omega_x - \theta_1$ and $\theta_L - \omega_y$ respectively.

Establishing an analytic solution for a prototype filter in the manner of Fig. 3.11 with a phase response of equation 3.26 is very difficult. Thus, as a beginning, the arguments of section 3.3.1 that led to an estimate of a 3x increase in impulse response length for a sine phase deviation are used here and suggests $N_{nlp} = 219$. Fig. 3.14a shows the phase specification equivalent to Fig. 3.10 but with the $\sin^2()$ fit and option (a) of equation (3.20a) used. Fig. 3.14b is the resulting passband. Compared with Fig. 3.12, the passband is far better controlled but still retains three large dips at the normalised frequencies of 0.00875, 0.044 and 0.06875 (70, 350, 550 Hz). The dip at 70 Hz defies explanantion, but those at 350 and 550 correspond to the phase transitions between the 3,4th and 5,6th phase intervals of Fig. 3.11. These phase transitions and particularly the one between the 5,6th interval exceed π according to equation (3.22a).

Where the assignment of the random sequence is by equation (3.20b) (option (b)), the final phase value at ω_y is the nearest multiple of 2π to $\theta(\omega)_L$ so that the situation of equation (3.22a) is avoided. Fig 3.15 shows marked improvement in the passband behaviour when option (b) is used. No phase plots are shown for either case because the phase response achieved was essentially indistinguishable from the specification.



(a) phase response using $\sin^2()$ fit and equation (3.20a)



(b) passband response.

Fig. 3.14: Passband and phase response achieved using Algazi's method on the example of Fig. 3.10 where the $\sin^2()$ fit has been used. N=219.

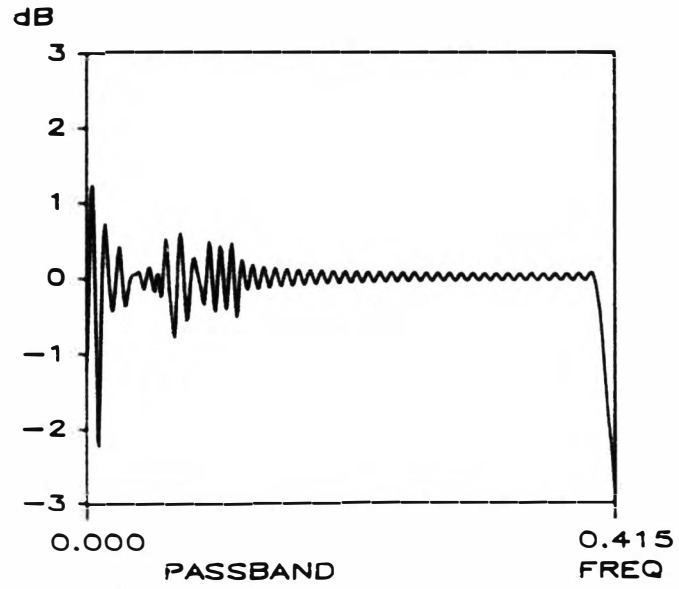


Fig. 3.15: Passband response achieved for the same example as Fig. 3.14 except that equation (3.20b) is used here. N=219

3.7 SUMMARY

The design of a non-linear phase filter has been shown to be a more difficult design problem because of the requirement to satisfy two constraints at the same time. For reasons not properly established, the algorithm of McLellan-Parks failed when given a non-linear phase problem.

The examples of the behaviour of Cuthbert's and Algazi-Suk's algorithms deliberately used the same impulse response length as Chapter 2 to retain the same common base for comparison. In doing so, it is apparent that Algazi's method generates the identical phase and passband response as Cuthbert's, but gives a poorer stopband result. However, Cuthbert's method has been shown to require a very large amount of computer time to provide a solution, in contrast to Algazi's method.

The iterative procedure of Algazi-Suk has been demonstrated to be useful in the presence of a non-linear phase with the same limitation that it apparently requires a longer impulse response length than other methods if the desired stopband specification is to be met.

Bearing in mind its simplicity of implementation, its relatively fast solution time and that it requires minimal additional array storage, this algorithm is considered to be an extremely useful method for situations that require a quick design to test an idea and where optimality is not an issue. Essentially, this method can be considered as the "general purpose" non-linear phase design method in the same way that the windowing method is regarded for the linear phase case. The discussion involving two opposite extremes of phase response has shown that in theory, a longer impulse response compared to the linear phase case will be required. In practice, however, the extra increase in length actually required will depend entirely on the form of the phase response. Because the required phase response will alter according to different users and needs, finding an analytic solution for every phase response is impractical, and in some cases so difficult that the only practical method is to compute the impulse response via an FFT. Thus, in practice, it will only be possible to estimate the required impulse response length if the phase response is simple enough to allow a straightforward analytic solution. This inevitably means that designing a non-linear phase filter to achieve desired magnitude and phase specifications will invlove more trial and error.

The discussion of the random phase filters showed the additional difficulties involved in achieving a practically useful filter. With the design methods available, the greatest difficulty lies in achieving an acceptably low passband ripple. In chapter 5.0, all the non-linear phase responses will be formed using the $\sin^2()$ fit.

CHAPTER 4.0

THE SPEECH SIGNAL

4.1 INTRODUCTION

As a pre-requisite to discussing the phase manipulation of speech, some understanding and knowledge of the speech signal and its underlying properties is required. This is the purpose of this chapter.

The material is a heavily condensed precis of the considerable body of knowledge that exists. The approach taken here is a descriptive one to elucidate the essential characteristics of the speech signal. The chapter concludes with the introduction of the Analytic vector and zero distribution representations of the signal because the work of Chapter 5 is built on these representations.

4.2 SPEECH PRODUCTION

The Human vocal tract mechanism responsible for the sounds that constitute speech is shown in Fig. 4.1. A mechanical model of the essential elements of the biological system is shown in Fig. 4.2 and forms the basis for the electrical and mathematical modelling of the production mechanism.

The energy source for speech production is the muscle groups of the chest and abdomen that force the air in the lungs through these components at a steady pressure (ie volume flow). There are three classes of excitation of the vocal tract:

(a) periodic

The vocal cords vibrate producing quasi-periodic pulses of air that excite the acoustic system above. The voice "pitch" is the fundamental frequency of this vibration. The waveform is roughly triangular as shown in Fig. 4.3 and thus contains many harmonics whose amplitudes decrease at about 12 dB/octave. This type of excitation is the basis for the formation of voiced sounds.

(b) Incoherent

This form of excitation is turbulent flow of air created at some point of constriction in the vocal tract. The turbulent flow creates an acoustic noise which is thus an incoherent form of excitation. This type of excitation forms the basis of unvoiced sounds.

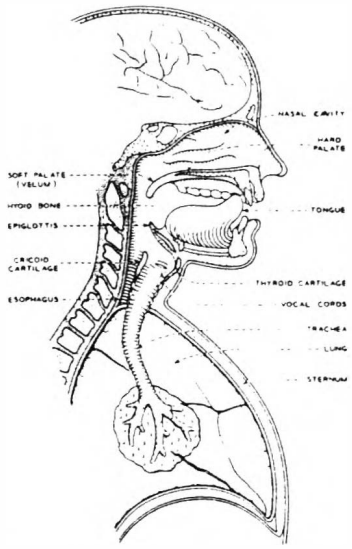


Fig. 4.1: The Human vocal mechanism
[after Flanagan, 1972]

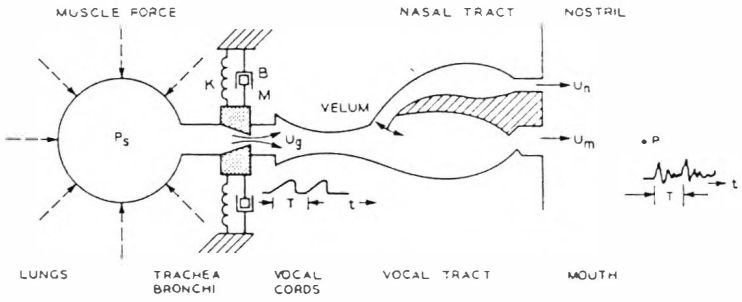


Fig. 4.2: Mechanical model of the human vocal mechanism
[after Flanagan, 1972]

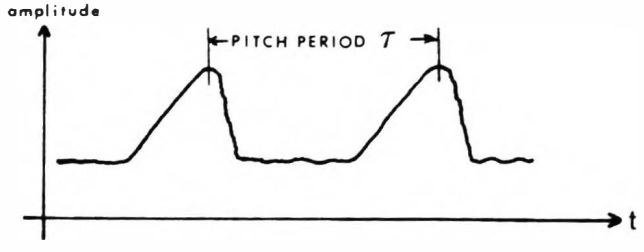


Fig. 4.3: The glottal pressure waveform for
voiced speech.

(c) Transient

Transient excitation is created by the build-up of pressure at some point of closure and then abruptly released. This results in a pulse (of air) to excite the acoustic system and is usually modelled as a step function of air pressure. This form of excitation is the basis of the plosive sounds in speech and can be used with or without vocal cord vibration, so the plosives can be voiced or unvoiced. For example /b/ from "bat" and /t/ from "tin".

4.3 ESSENTIAL PROPERTIES

There are many ways of describing the speech signal s(t), each of which contribute some aspect towards understanding it. The speech can be described from a linguistic viewpoint with the grammatical rules of the language as constraints. When strung together, these linguistic elements produce a signal which can be analysed and objective measurements made on the waveform. This gives an understanding of the underlying physical structure of the speech signal. Subjective measurements (opinions) based on experiments involving hearing can also be made.

The description of the essential properties below is confined principally to the physical structure of the speech signal.

4.3.1 Phonemes

The three classes of excitation described above produce different sounds. A particular set of combinations of these sounds produces the basic "code symbols" that constitute the linguistic elements of a language. These code symbols are known as "phonemes". However, the shape and size of the cavities in the vocal system can be altered in a complex manner, so that the same phoneme can sound different depending on the context and the emphasis placed on it by the speaker. These acoustic differences of the same code symbol are referred to as "allophones".

The principal groups of the phonemes in the English language are shown in Fig. 4.4. Each sound, when considered in isolation, can be classified as either continuant or non-continuant sounds. Continuant sounds are produced by a fixed, non-time-varying, vocal tract configuration stimulated by the appropriate excitation source.

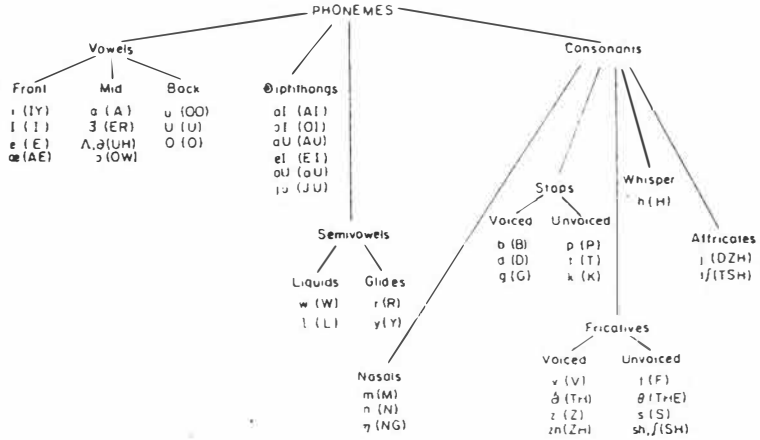


Fig. 4.4: A phonetic and orthographic representation of the phonemes of American English. [after Rabiner et al, 1978]

Non-continuant sounds are produced by a changing vocal tract configuration.

In conversational speech, the vocal tract is in constant motion and alters in a complex manner to set up at each instant the necessary conditions for each sound. Sometimes, the vocal tract changes its configuration during the utterance of a sound. At other times, there is a very short pause called an "inter-phoneme silence" during which the vocal tract alters its configuration before the next sound is produced. Such very small pauses are usually ignored (or are indistinguishable) when performing an analyis of the speech signal. For this reason, the most useful partitioning of speech is in terms of voiced and unvoiced sounds. This partitioning, in conjunction with the two basic types of excitation (periodic and incoherent) leads to the widely used general electrical model for speech production shown in Fig 4.5.

4.3.2 Time Domain

From Fig 4.5, it is apparent that speech is a non-stationary signal; that is, because speech is produced largley by two sources whose characteristics and properties are almost entirely opposite, the characteristics of the speech signal will alter with time. This explains why it is usual to consider speech as voiced or unvoiced. The only other method available which gives a relatively stable measure of

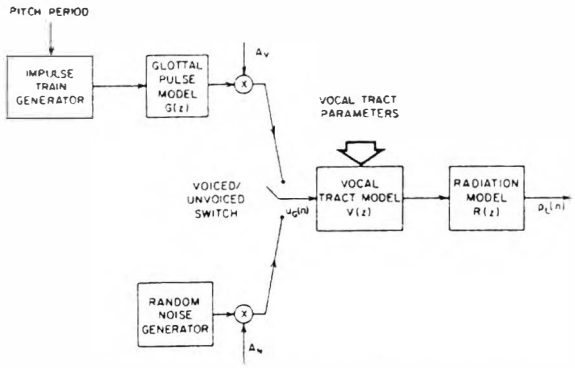


Fig. 4.5: Electrical model of the speech production mechanism. [after Rabiner et al, 1978]

the time waveform is a statistical description.

Davenport [1952] studied the signal amplitude probability distribution and established an empirical relation describing the distribution. Later, Paez [Paez et al, 1972] established that the Gamma distribution was a very close fit to the real speech distribution as shown in Fig 4.6. This shows that speech has a very high probability of low signal amplitudes and a small but finite probability of large signal amplitudes.

The signal amplitude probability distribution indicates that the speech signal has a large dynamic range which poses problems in electronic communications systems. Indeed, Purton [1962], in discussing a choice of PCM companding law indicated that a total dynamic range of 62 dB was theoretically required, but that a range of 38 dB is satisfactory to accommodate 99.95% of all conversations.

The non-stationary nature of the speech signal causes considerable problems when attempting to define and measure the signal power. Because of the complexity of the signal, the traditional measures of rms and peak signal values as indicators of signal power are difficult to apply in a meaningful way because they are changing continuously.

This problem was recognised in the early years of electronic communications systems and led to the introduction of another unit called the "volume unit". This unit of measurement is related to the loudness of the signal and relies on two aspects when indicating a

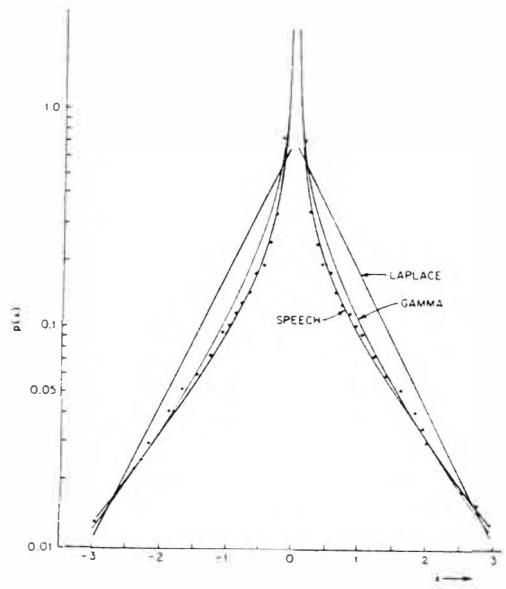


Fig. 4.6: Probability density function of real speech compared with the theoretical Gamma and Laplace probability densities. [after Paez et al, 1972]

value: an electrical reference level relative to an rms value of a sinewave and the dynamic characteristics of the instrument itself in terms of internal electrical time-constants and the ballistics of the meter movement [Chinn et al, 1940]. However, this approach merely sidesteps the basic problem of establishing a useful measure of the long term power level of the signal that is easily identified with some common, well recognised unit of measure such as the dBm.

From the point of view of preventing saturation in electronic circuits within a communications environment, knowledge of the real signal peaks is desirable. Brady [1965] studied this problem using the probability distribution of speech as a base. He showed that speech has a largely log-uniform distribution, so that a measure he called "Average Peak Level" (APL), computed from the time average of the log of the envelope waveform will give a result that is the same as the actual peak of the signal if it could be measured. Later, he defined an "Equivalent Peak Level" (EPL) [Brady, 1968] which is based on an rms measurement of the signal. This EPL measure was an improvement over the APL measure because the EPL matches changes in speech level on a dB-dB basis over a 35 dB range.

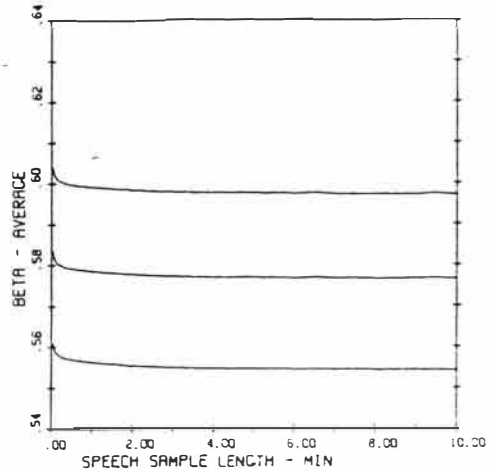


Fig. 4.7: Speech sample length versus the ratio of absolute average magnitude to rms level. [after Niederjohn et al, 1983]

In related work, Neiderjohn et al [1983] has shown that for high quality speech (wide bandwidth, low noise) utterances longer than about 20 seconds, the ratio of average absolute magnitude to rms level of the speech is almost constant at 0.55 for males and 0.6 for females as shown in Fig. 4.7.

4.3.3. Frequency Domain

The time varying nature of speech leads to problems in establishing the spectral charactersitics of the signal as a whole. Simply taking the Fourier transform of any given segment of s(t), of arbritrary (time) length and using the result as an adequate description of the spectral properties of the signal is not sufficient. This problem has led to the development of the "Short-time" Fourier transform (STFT) which has some important consequences, especially for applications related to reducing the bit rate required for speech coding systems.

It has been demonstrated that the STFT can be considered from two different aspects. The first is that it can be shown that the STFT is equivalent to splitting the composite speech signal into separate frequency bands by a bank of bandpass filters. The other approach is that it can be considered as a normal Fourier transform that gives the

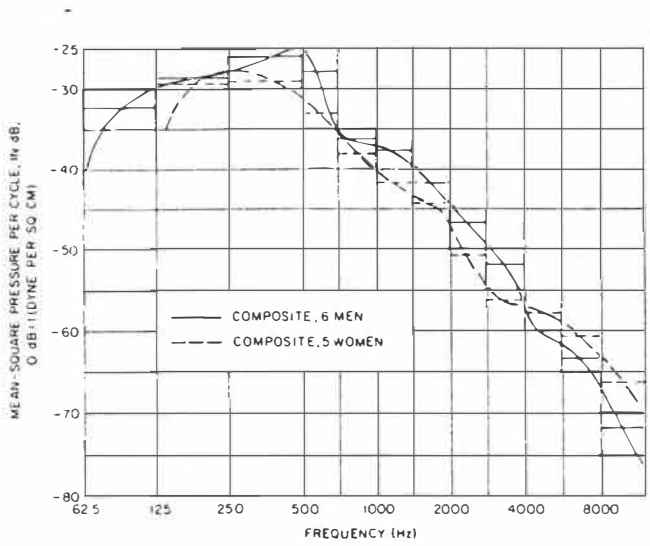


Fig. 4.8: Long term power spectral density of continuous speech. [after Dunn et al, 1940]

spectra of the short time segment selected. [Schafer et al, 1973; Schafer et al, 1974; Allen et al, 1977]

The long time power density spectrum of speech is shown in Fig. 4.8. and indicates that most of the speech energy lies below 750 Hz.

This region corresponds to voiced speech which exhibits strong periodicity and high amplitudes. However, it has already been pointed out that conversational speech requires a constantly changing vocal tract configuration. Thus, while a single voiced sound taken in isolation may be considered to have a fixed amplitude and harmonic structure, conversational speech will have a continuously changing spectral structure. This implies that the fundamental frequency of voiced speech is not constant as shown in Fig 4.9 [Ross et al, 1974; Gold et al, 1969b]

Thus, it is more usual to regard the harmonic structure of voiced speech as "teeth" of finite width centred on some average spacing as shown in Fig 4.10. The reciprocal of this "average spacing" is known as the "pitch period". Such a practice is a gross over-simplification but practically useful.

The pitch period also varies between speakers of the same sex, with large differences between men, women and children. Vivalda et al [1984] performed a statistical analysis of the pitches of 112 voices.

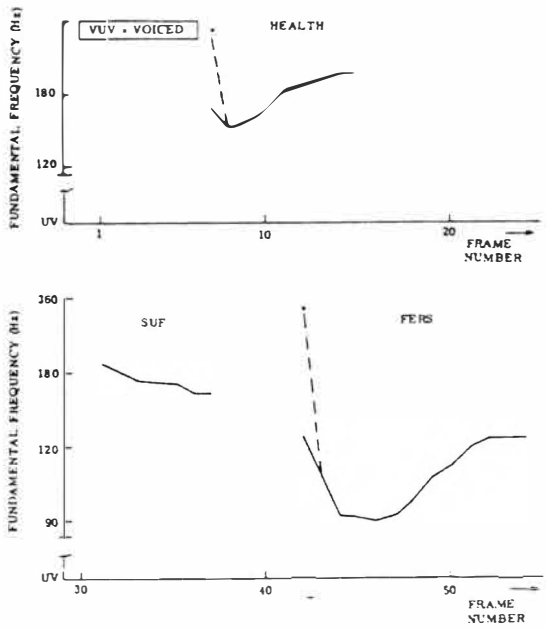


Fig. 4.9: Variation in Fundamental frequency with utterance.
[after Ross et al, 1974]

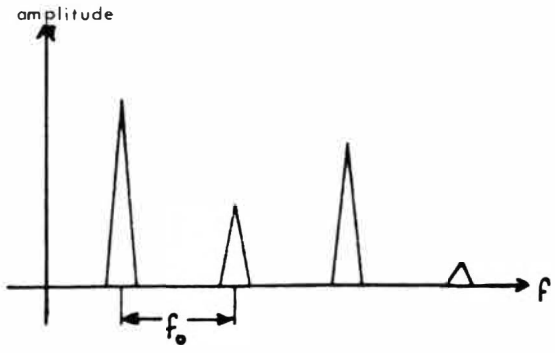


Fig. 4.10: Representation of the harmonic structure of speech.

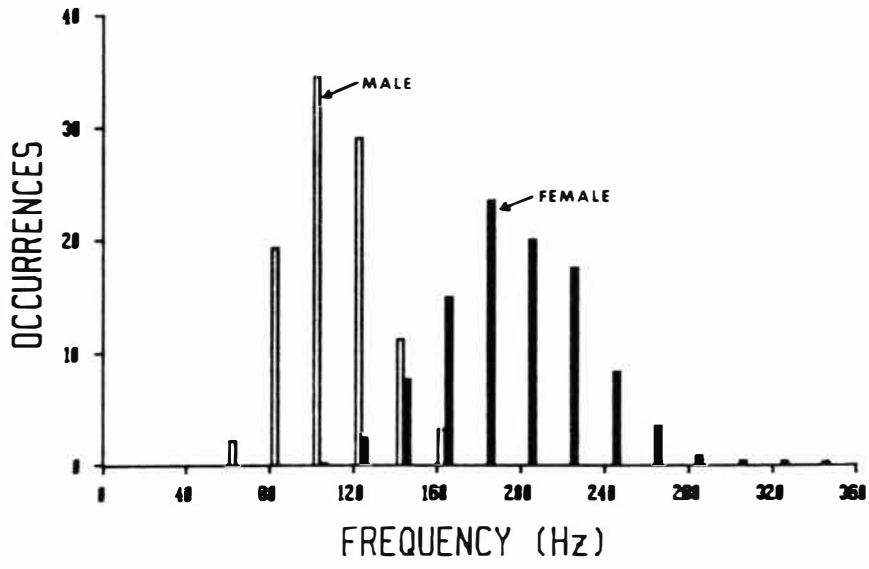


Fig. 4.11: Pitch (fundamental frequency) distribution of 64 male and 48 female voices. [after Vivalda et al, 1984]

Their results as shown in Fig. 4.11 indicate that, as general rule-of-thumb, the pitch period of a female voice is twice that of a male.

The amplitude structure of the speech we hear is modified by the resonances of the vocal tract. The peaks in the envelope of the signal reflect these resonances which are referred to as the "formant frequencies". These formants are not stable, but change with different utterances [Schafer et al, 1969], and to a lesser extent between speakers for the same utterance.

In contrast to voiced speech, unvoiced speech stemming from an incoherent source has spectral characteristics of noise modified by the formants of the vocal tract. Energy from unvoiced speech is very low and contains no periodic structure. Most of the energy is distributed in the higher frequencies. Fig 4.12 shows the spectral models for voiced and unvoiced speech with the formant structure evident. Taken together and averaged over many speakers and utterances results in the long time power density spectrum of Fig 4.8.

4.4 PERCEPTION

The process of hearing involves the ear and the brain. The ear is

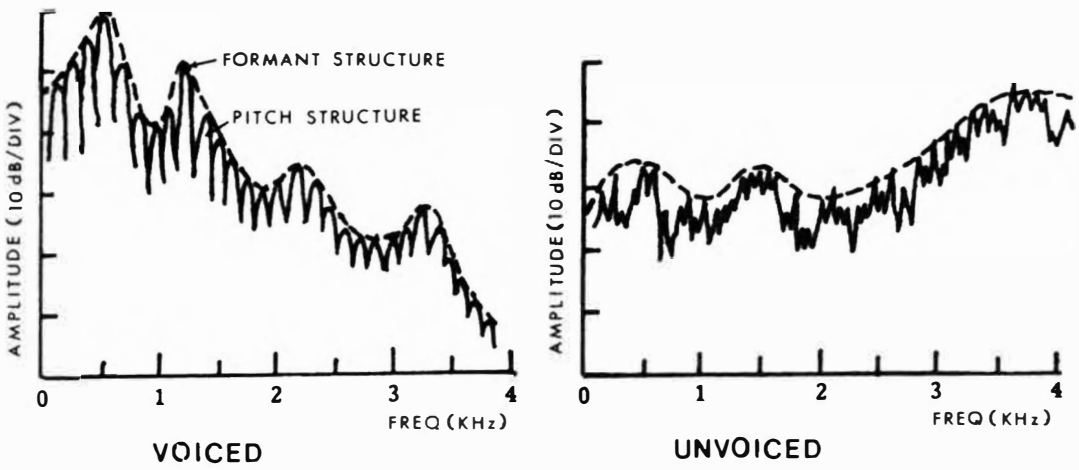


Fig. 4.12: Formant structure of voiced and unvoiced speech. [after Crochiere et al, 1983]

the mechanism for converting the acoustic signal to neural signals which are then processed by the brain. The details of the arguments relating to the mechanism of hearing are not at issue in this work. The important aspect here is that the process of hearing allows subjective measures of the speech signal to be made that complement the objective measures. Thus any discussion on speech usually involves the hearing process and the opinions formed.

4.5 JUDGING SPEECH

The objective measurements on the speech signal itself provide quantitative information about the signal. Carefully controlled experiments involving the hearing process give qualitative information that complements the objective measurements. The basic information in the speech signal is the words formed by the vocal mechanism within the grammatical rules of the language. It has been established that the ear performs a rough frequency analysis of the acoustic signal and that it is the short-time amplitude spectrum that is the dominant influence on speech intelligibility.

Speech intelligibility is a subjective measure of the clarity and unambiguity of the basic message carried by the language. The spectra and time behaviour of the signal are easily quantified. However, two signals with the same spectrum can sound completely different, simply by altering the phase relationship of the components [Plomp et al,

1969]. Thus the speech signal contains additional information that is not readily apparent. This additional information is described by the word "quality".

The quality of a speech signal is also a subjective measure. It is manifested and controlled by such factors as speaker identification, emotion, unusual or different emphasis in the vocalisation. These factors are outside the grammatical rules of the language, but contribute a great deal to the information transmitted. This subjective information is extracted by the correlative abilities of the brain.

It is this extra information that creates problems in speech communications systems that seek to minimise the information rate (ie bandwidth) needed to transmit the basic message. For telephone bandwidth speech, "toll quality speech" in a communications system gives low distortion and high signal-to-noise ratio for the basic message while retaining enough of the subjective information that a listener perceives the received speech to be the same as if hearing the speaker directly. As more of the side information is lost with possibly increased distortion of the basic message, the speech is perceived as losing quality and may be referred to as "communications quality". The opposite extreme of vocoder speech (synthetic speech) is subjectively identified as being of "mechanical quality" because while the basic message of the language is sufficiently intelligible that it is understood unambiguously, it requires additional effort on the part of the listener to decode the message because the subjective information is missing.

Effects due to the speaker's physical condition, such as suffering from a common cold, are not part of the "quality" description because these effects are quantifiable as part of the physical structure of the waveform: the vocal tract and its resonances are physically modified. In the case of a cold, the nasal cavities may be blocked and this eliminates the radiation from the nostrils as well as modifying the formant structure.

The foregoing discussion raises an important point that is often not well appreciated. This is that the subjective terms "intelligibility" and "quality" refer to two entirely different but related aspects of the speech signal. Intelligibility, as already mentioned, is a measure of the amount of distortion of the basic

message carried by the language. High intelligibility implies little distortion of the basic message and is thus easy to decode. The term "quality" relates to the side information which modifies the basic structure of the signal in a subtle manner that the correlative abilities of the brain can interpret to give contextual information and speaker identification. This "side information" is not part of the basic message.

4.6 MATHEMATICAL REPRESENTATION OF VOICED SPEECH

The discussion so far has been descriptive and was intended so to give a broad understanding of what speech is made up of and how it can be described. Now, two mathematical representations which can be applied to voiced speech are introduced and illustrated because they will constitute the basis on which the arguments of Chapter 5.0 are built.

Voiced speech has been shown to be strongly periodic with large signal amplitudes. Over short time intervals, typically two to three pitch periods of a male voice, the structure of a voiced sound is essentially constant. The traditional means of describing the physical attributes of such a signal are well understood (section 4.3.2 and 4.3.3).

The mathematical tools that will be used in addition to Fourier Analysis via the FFT are the Analytic Vector Representation and Zero Distribution. These representations were pioneered by Voelker [Voelker, 1966a,1966b]. The reason for using this representation here is that the Analytic vector $\psi(t)$ gives a physical picture of the signal s(t) that shows much more clearly the effect of manipulating some aspects (the zeros) of the signal. Furthermore, by adopting the Analytic representation, the number of zeros required to describe the signal is halved.

Hamilton [Hamilton,1985] has made an extensive study of the Analytic Vector Representation of speech, and his work provides additional insights into the characteristics of speech.

The following two sections outline the bare bones of the theory. Subsequently, some examples give physical meaning to the concepts and serve as a useful vehicle to expand on some additional aspects of the basic theory.

4.6.1 Analytic Vector Representation

An analytic signal $\psi(t)$ is defined as

$$\psi(t) = s(t) + j\hat{s}(t) \tag{4.1}$$

where $s(t)$ is the real signal and $\hat{s}(t)$ its Hilbert transform defined by

$$\hat{s}(t) = \frac{1}{\pi}\int_{-\infty}^{\infty} \frac{s(t)}{t-\tau}\, d\tau \tag{4.2}$$

The analytic signal of equation (4.1) is complex, so it can be written in terms of its magnitude and phase:

$$\psi(t) = m(t)e^{j\phi(t)} \tag{4.3}$$

where

$$m(t) = \left[s^2(t) + \hat{s}^2(t)\right]^{\frac{1}{2}} \tag{4.4}$$

$$\phi(t) = \tan^{-1}\left[\hat{s}(t)/s(t)\right] \tag{4.5}$$

$m(t)$ is the instantaneous envelope and $\phi(t)$ the instantaneous phase of the real signal since

$$\begin{aligned} s(t) &= \mathrm{Re}[\psi(t)] \\ &= m(t)\cos\phi(t) \end{aligned} \tag{4.6}$$

If $S(\omega)$, $\hat{S}(\omega)$, $\Psi(\omega)$ are the Fourier transforms of $s(t)$, $\hat{s}(t)$ and $\psi(t)$ respectively, it is known that

$$\hat{S}(\omega) = -j\,\mathrm{sgn}(\omega)S(\omega) \tag{4.7}$$

$$\text{where } \mathrm{sgn}(\omega) = \begin{cases} 1, & \omega > 0 \\ 0, & \omega = 0 \\ -1, & \omega < 0 \end{cases}$$

so that the Fourier transform of equation (4.1) is

$$\Psi(\omega) = \begin{cases} 2S(\omega), & \omega > 0 \\ S(0), & \omega = 0 \\ 0, & \omega < 0 \end{cases} \tag{4.8}$$

$\Psi(\omega)$ is single sided and thus $\psi(t)$ only occupies half the bandwidth of $s(t)$. Fig. 4.13 illustrates the signal $s(t)$ and envelope $m(t)$ of a

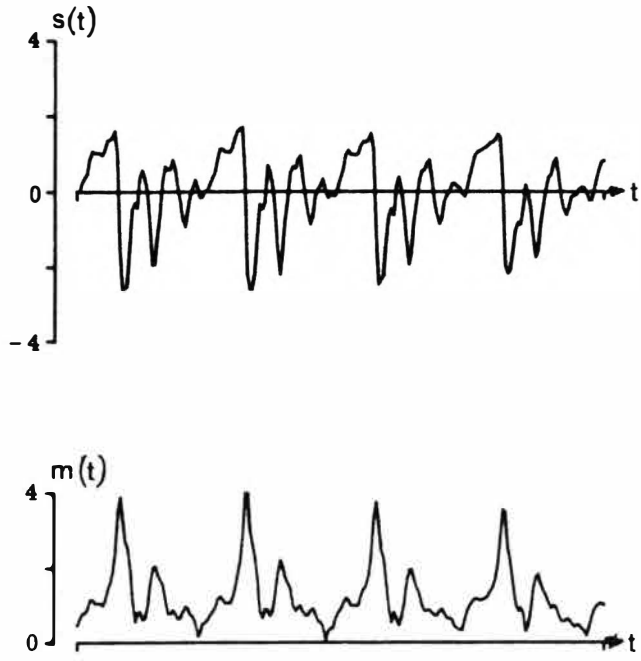


Fig. 4.13: 3 cycles of the signal s(t) and its envelope m(t) of a real vowel.

real speech signal.

4.6.2 Zero Representation

It is well known that if a signal s(t) is periodic in $T=2\pi/\Omega$ and bandlimited to $\pm W=n\Omega/2\pi$, then the signal can be represented by a finite Fourier series

$$s(t) = \sum_{k=-n}^{n} c_k e^{jk\Omega t} \qquad (4.9)$$

where the c_k are the complex Fourier coefficients representing the amplitude and phase of each sinusoid.
If the time variable t is allowed to be complex

$$t \rightarrow z = \tau + j\sigma \qquad (4.10)$$

then the function s(z) is completely determined by the principal strip of the complex plane shown in Fig. 4.14.
Using the mapping

$$x = e^{j\Omega z} \qquad (4.11)$$

and equation (4.10) in (4.9), then

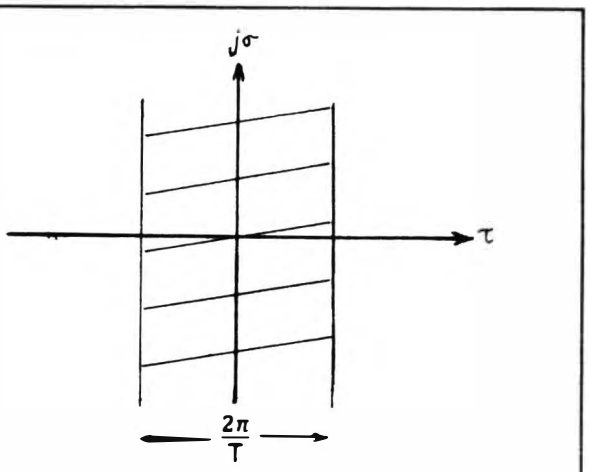


Fig. 4.14: Principle interval of the complex z-plane.

$$s(z) = \sum_{k=-n}^{n} c_k x^k \tag{4.12}$$

which is now an algebraic polynomial.The mapping of equation (4.11) takes the upper half plane (UHP) of the principal strip onto the interior of the disk $|z|=1$; the lower half plane (LHP) of the strip onto the exterior of the disk and the real τ axis onto the circle $|z|=1$. Solution of the equation

$$\sum_{k=-n}^{n} c_k x^k = 0 \tag{4.13}$$

obtains a set of zeros $\{x_k\}$ of the polynomial(and thus of the signal s(t)). If, in addition, s(t) is real (ie physically realisable), then basic Fourier theory requires the Fourier coefficients c_k exhibit Hermitian symmetry. Thus all the zeros of s(z) are either real or occur as complex conjugate pairs.

The order of equation (4.13) is 2n which is always even. Recalling the opening statement of this section, this means that a real signal, bandlimited to $n\Omega/2\pi$ has an even number 2n zeros per period of the z-plane.

The positions of the roots on the complex plane are given by

$$x_k = e^{j\Omega z_k}, \qquad k=1,\ldots,n \tag{4.14}$$

and taking non-principal logarithms gives

$$\ln(x_k) = \ln|x_k| + j(2\pi\ell + \mathrm{Arg}[x_k]) \qquad (4.15)$$

whence

$$z_k = \tau_k + j\sigma_k = \frac{1}{\Omega}(2\pi\ell + \mathrm{Arg}(x_k)) - \frac{j}{\Omega}\ln|x_k| \qquad (4.16)$$

where $2\pi\ell$ are the repeated periods of the principal interval of Fig. 4.14.

The reader is reminded at this point that the discussion above has described two different, but closely related, sets of zeros. The sequence of zeros $\{x_k\}$ are the roots of the polynomial of equation (4.13). The zeros $\{z_k\}$ are related to $\{x_k\}$ through the mapping of equation (4.11) and hence equation (4.16). These $\{z_k\}$ will be referred to as "analytic zeros". Section 4.6.4 that presents some examples will show that the locations of these $\{z_k\}$ correspond to clearly identifiable features of the analytic envelope m(t).

Relating the foregoing arguments to the analytic signal of equation (4.1), $\psi(t)$ has a single sided spectrum. Thus, the equivalent equation to equation (4.13) for $\psi(t)$ is

$$\psi(z) = \sum_{k=0}^{n} c_k x^k \qquad (4.17)$$

which is a polynomial of order n+1 and therefore must possess n zeros.

These zeros (the $\{x_k\}$) are all complex and occur singly - that is, they do not occur as complex conjugate pairs. This is because to get a real zero requires m(t)=0 in equation (4.4). This in turn requires both s(t) and $\hat{s}(t)$ be zero simultaneously. For a periodic sequence containing more than one harmonic (ie n>1 in equation (4.9)), the practical effect of the Hilbert transform is to alter the phase of each component of $\hat{s}(t)$ by $\pi/2$ from s(t). Under these conditions, it is physically impossible for both s(t) and $\hat{s}(t)$ to be zero simultaneously. Thus, all the zeros are complex.

The sequence of zeros $\{x_k\}$ (and thus $\{z_k\}$) are informational attributes of the signal s(t) in the same way that the Fourier coefficients c_k are. Since the polynomial of equation (4.13) has 2n roots, then it can be equivalently represented by a product expansion of the roots [Requicha, 1980]

$$s(z) = c_n \prod_{k=-n}^{n} (x-x_k)$$

$$= \sum_{\substack{k=-n \\ k\neq 0}}^{n} c_k x^k \qquad (4.18)$$

Voelker, in examining the analytic parameters of a signal, showed that the signal can be expressed as two product terms

$$s(t) = s_{RZ}(t).s_{CZ}(t) \qquad (4.19)$$

where $s_{RZ}(t)$ and $s_{CZ}(t)$ are the factors that contain only real or complex zeros. Computing the analytic zeros $\{z_k\}$, he showed that (ignoring scaling constants)

$$s_{RZ}(t) = \prod_{(k,\ell)=1}^{n_R} \left[\cos\Omega\left(\frac{\tau_k - \tau_\ell}{2}\right) - \cos\Omega\left(t - \frac{\tau_k - \tau_\ell}{2}\right)\right] \qquad (4.20)$$

$$s_{CZ}(t) = \prod_{i=1}^{n_C} \left[\cosh\Omega|\sigma_i| - \cos\Omega(t-\tau_i)\right] \qquad (4.21)$$

where the $2n_R$ real zeros are taken as n_R pairs (x_k, x_ℓ). He further showed that instantaneous envelope

$$m(t) = s_{CZ}(t) \qquad (4.22)$$

Thus the instantaneous envelope (and phase) fluctuations of the analytic signal can be generated from its set of analytic zero locations. For the envelope, the occurrence of each zero z_k causes a dip in the envelope at time τ_k. The intensity of the dip k is dependent on the magnitude of the imaginary coefficient of z_k and becomes greater as $|\sigma_k| \to 0$ [Hamilton, 1985].

In addition, the Analytic vector representation of section 4.6.1 shows that $\psi(t)$ contains a complete description of s(t), but $\psi(t)$ possesses only half as many zeros. The implication is that since no information is lost, each complex conjugate pair of zeros of s(t) collapse to become one complex zero of $\psi(t)$.

4.6.3 Computing the Zeros

In order to extract the zeros $\{z_k\}$ of the polynomial of equation

(4.13), the Fourier coefficients c_k must be determined first. This would normally be done by using an FFT algorithm based on a highly composite number 2^N, whose initial data is a segment of s(t), windowed by a suitable window function to minimise spectral leakage [Harris, 1978]. In this case, this technique is not possible, because, in the manner of equation (2.21), the process of windowing gives

$$s(t)w(t) \quad \langle \xrightarrow{FT} \rangle \quad S(\omega)*W(\omega) = S_r(\omega)*W(\omega) + jS_q(\omega)*W(\omega) \tag{4.23}$$

where $\langle \xrightarrow{FT} \rangle$ means Fourier transform and * is the convolution operator. The Fourier phase $\theta(\omega)$

$$\theta(\omega) = \tan^{-1}\left(\frac{S_r(\omega)*W(\omega)}{S_q(\omega)*W(\omega)}\right) \tag{4.24}$$

has the spectrum of the window function embedded. Thus, applying a window to the data segment hides the true value of $Arg[c_k]$ of equations (4.13) and (4.17). For this reason, either an explicit DFT must be performed, or a mixed radix FFT algorithm such MXFFT [Singleton, 1979] must be used. For the work reported here, Singleton's algorithm MXFFT is used to extract the c_k.

Subsequently, because the zeros of the analytic signal $\psi(t)$ are required, the conditions of equation (4.8) are first applied to the c_k before they are used with an algorithm to extract the zeros. For this work, the program CPOLY [Jenkins et al, 1972] is used. This algorithm is organised to find the zeros of the polynomial

$$\sum_{k=0}^{n} \rho_k z^{n-k} = 0 \ , \quad \rho_0 = 1, \quad \rho_n \neq 0 \tag{4.25}$$

where $\rho_0 = 1$ is for convenience only. This order of coefficients is the reverse of equation (4.13): ie ρ_n in equation (4.25) corresponds to c_0 of equation (4.13). For bandpass signals, $c_0 = 0$ which contravenes the requirement above. In this case, the term c_0 is ignored and the degree of the polynomial is reduced by 1 at the time CPOLY is invoked.

For convenience and to control the magnitudes of the coefficients given to CPOLY, the ρ_k are normalised to ρ_0. The connection between equation (4.25) and (4.13), using equation (4.8) is

$$\rho_k = 2c_{n-k}/2c_n \quad , k = 1,..n-1$$
$$\rho_n = c_0/2c_n \tag{4.26}$$

so that $\rho_0 = 1$.

4.6.4 Examples illustrating the theory

4.6.4.1 Real signal s(t)

In the manner of Voelker, consider the signal

$$s(t) = A + a\cos\Omega t \tag{4.27}$$

Using equations (4.10) and (4.11),

$$s(z) = A + \frac{a}{2}\left(x + \frac{1}{x}\right)$$
$$= \frac{a}{2x}\left(x^2 + \frac{2Ax}{a} + 1\right) \tag{4.28}$$

which has roots at

$$x = \frac{1}{a}\left(-A \pm \sqrt{(A^2 - a^2)}\right) \tag{4.29}$$

Fig 4.15 shows equation (4.27) and the zero positions. For A=0, s(t) is a simple sinusoid with no DC offset and x in equation (4.29) is pure imaginary ($\sqrt{-a^2}$). Using equation (4.27) in (4.16) gives

$$\tau = \frac{Arg(x)}{\Omega} = \frac{\pm\pi/2}{\Omega} \tag{4.30a}$$

$$\sigma = -\frac{1}{\Omega}\ln|x| = -\frac{1}{\Omega}\ln(1) = 0 \tag{4.30b}$$

The real time locations τ are simply the positions of the zero crossings of s(t).

For A<a (A>0), the zeros remain real but begin to bunch together in pairs. At A=a, the real first order zeros merge to become real second order zeros. Then, as A>a, these second order real zeros split into a pair of complex conjugate zeros.

For this trivial example with only one harmonic, it is easy to relate the analytic zero positions to features of the signal waveforms. However, for a signal with many harmonics, it would be very difficult to relate the zero positions to features of the signal waveform.

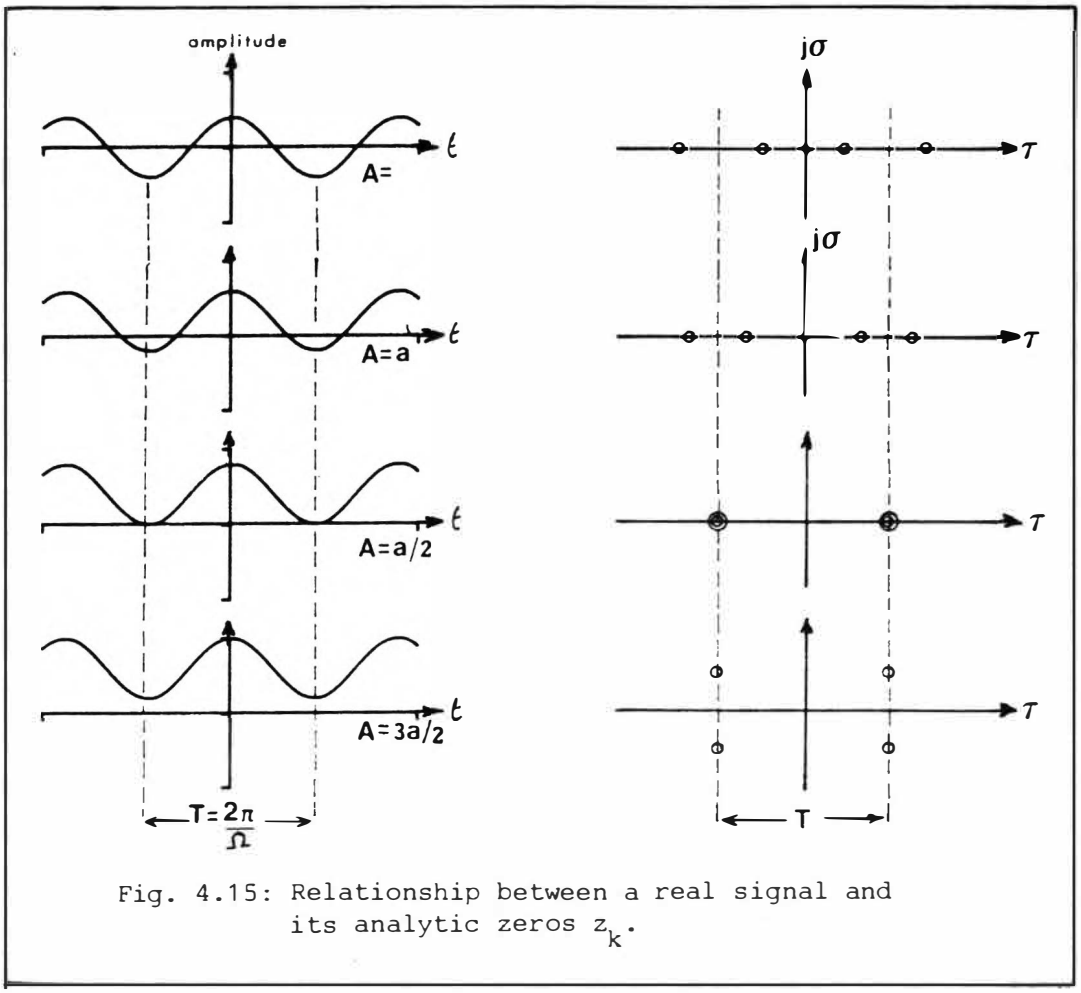


Fig. 4.15: Relationship between a real signal and its analytic zeros z_k.

4.6.4.2 Analytic signal $\psi(t)$

Using s(t) of equation (4.27), from equation (4.7), the Hilbert transform is non-constant preserving, so the Hilbert transform of s(t) is

$$\hat{s}(t) = a\sin\Omega t \tag{4.31}$$

Thus,

$$m(t) = \left((A+a\cos\Omega t)^2 + a^2\sin^2\Omega t\right)^{\frac{1}{2}}$$

$$= \left(A^2 + a^2 + 2Aa\cos\Omega t\right)^{\frac{1}{2}} \tag{4.32}$$

and making the same substitutions used previously, equation (4.32) becomes

$$m(z) = \left(\frac{Aa}{x}\right)^{\frac{1}{2}} \left(x^2 + \frac{(A^2+a^2)}{Aa} + 1\right)^{\frac{1}{2}} \tag{4.33}$$

which has roots

$$x = -\frac{1}{2Aa}\left\{(A^2+a^2) \pm (A^2-a^2)\right\} \qquad (4.34a)$$

$$= -\frac{a}{A} \quad (+) \quad \text{or} \quad -\frac{A}{a} \quad (-) \qquad (4.34b)$$

In this case, x is purely real, so in equation (4.16), Arg(x)=0 and $\tau = 2\pi\ell/\Omega$. The case A=0 implies $x=-\infty$ or 0, which gives $|\sigma|=\infty$. A zero at ∞ is called a "removed" zero. This comes from a restatement of the fundamental theorem of algebra [Hille, 1959] that "every polynomial of degree ≥ 1 in z has a zero". The condition A=0 in equation (4.32) is equivalent to $\psi(z)=a$ which is a polynomial of order 0 - ie a constant.

Thus a "removed" zero is simply the reduction of the order of the polynomial by one. Equation (4.32) is of order 1 ($\cos\Omega t$), which, on reduction becomes order 0 or a constant.

The important point that comes out of the above discussion is that the original statement requiring equation (4.17) possess n zeros is that these n zeros must be non-removed zeros. Thus, the c_k are an explicit indication of the number of zeros in the (analytic) signal.

For $A\neq 0$, both solutions of equation (4.34b) are true when used in equation (4.33). These give

$$\tau = \ell\pi, \qquad \ell = 1,\ldots$$

$$\sigma = -\frac{1}{\Omega}\ln|a/A| \quad \text{or} \quad -\frac{1}{\Omega}\ln|A/a| \qquad (4.35)$$

Which particular solution is the correct choice depends on whether the signal is a minimum phase (MP) or non-minimum phase (NMP) signal. A minimum phase signal is one whose analytic zeros exist only in the lower half plane (LHP) [Voelker, 1966a]. In general, unless special measures are taken, a signal will be non-minimum phase and thus will possess zeros in both half planes. A simple way to ensure that a signal is MP is to add a DC bias A>0 to preclude the signal crossing the real time axis [Voelker 1966c]. In addition, Hamilton (1985) showed that the instantaneous frequency $\omega_i(t)$ defined as

$$\omega_i(t) = d\phi(t)/dt \qquad (4.36)$$

where $\phi(t)$ is the instantaneous phase of equation (4.3), exhibits large peaks or dips at the real time locations τ of the zeros and the direction of the spike indicates whether the zero is upper half plane

(UHP) or LHP. These spikes also occur at the same time as the dips in the envelope m(t).

For the trivial example above with one harmonic, the signal is maximum phase for A≤a which requires the zero in the UHP. When A>a, the signal becomes minimum phase and the zero is a LHP zero.

Fig. 4.16 shows the envelope and analytic zeros (of $\psi(t)$) corresponding to equation (4.32) for A=a/2 and A=a. Note that for both Figs. 4.15 and 4.16, the vertical scales for the time function and zero positions are different. Thus, for $|\sigma|>0$, there is no requirement that the zero position coincides with the dip of the time function. Note carefully in Fig. 4.16 that this trivial example containing only one harmonic is allowed to possess real (analytic) zeros for the case A=a because the envelope is allowed to touch the real axis.

The distinction between this example and the earlier one using the real signal is that when the signal contains many harmonics, the analytic zeros (computed from the signal) can be related to easily identifiable features in the envelope of the equivalent analytic

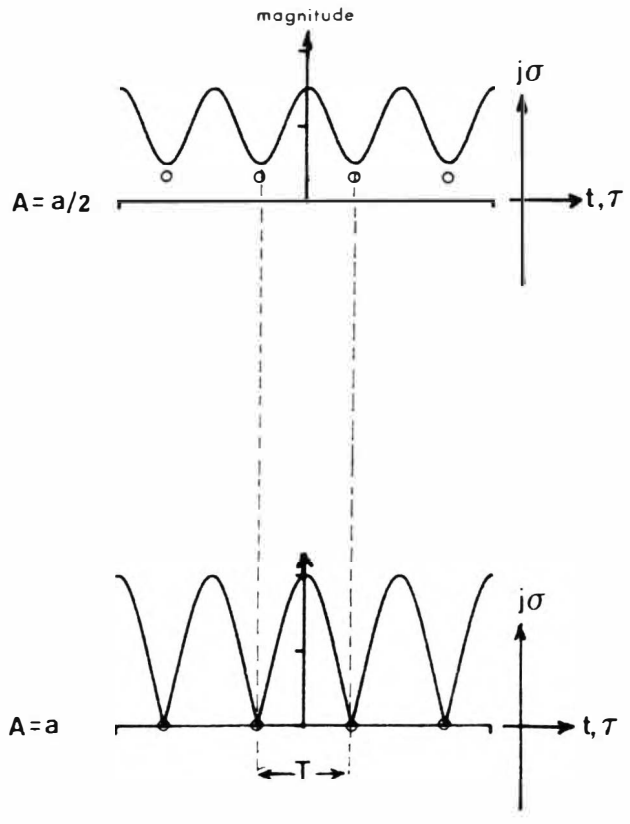


Fig. 4.16: Relationship between the envelope m(t) of an analytic signal and its analytic zeros z_k.

signal.

Finally, the two apparently different zero patterns of Fig. 4.15 and 4.16 are directly related, as indicated at the end of section 4.6.2. Here, the singly occurring complex zero of Fig. 4.16 corresponds to a complex conjugate pair of Fig. 4.15.

4.7 SUMMARY

This chapter has introduced the speech signal. A condensed overview of its production and properites has been given which has demonstrated that the behaviour of the speech signal is complex and requires several different tools to adequately describe it. Two mathematical representations of voiced speech have been introduced and described.

The Analytic envelope helps to give a physical picture of the signal behaviour. The concept of the zeros of a signal as the roots of a polynomial is well established, but bears little obvious relation to the original signal. This approach relies on the fact that voiced segments of speech are strongly periodic and can thus be modelled using a finite Fourier series. The analytic zeros pioneered by Voelker have been shown to be intimately related to clearly identifiable fluctuations of the analytic parameters of the signal. Since it is only half a step back to the original signal from the analytic representation, these analytic zeros are more easily comprehended in relation to the signal characteristics.

The issues related to the measurement of speech power remain the subject of continuing discussion and investigation. Today, the characteristics of all modern speech power measuring instruments are specified by international standards - eg IEC 651, 1979: Sound Level Meters. In addition, Campbell's work [1980] on developing a speech power measuring device highlights the number and complexity of the issues involved.

CHAPTER 5.0

SPEECH PHASE RANDOMISATION

5.1 INTRODUCTION

The previous chapter provided a base for describing and understanding the physical structure of the speech signal. Considered as a signal transmitting information, it is acknowledged today that the short-time amplitude spectrum is the dominant factor in establishing the intelligibility of the message, but there is significant subjective information in the signal the human brain can extract and is described by the word "quality". One aspect of the signal covered by the "quality" description is the Fourier phase spectrum of, in particular, voiced speech.

The part the short-time phase spectrum of speech plays in the perception and judgement of the quality of speech is not well understood. Schroeder [1975] asserts that if a short time segment of about 50 ms of voiced speech has its phase spectrum modified by Fourier transform, the acoustical quality of speech remains unaltered - ie the ear is "phase deaf". In addition, he has also said that phase manipulations that correspond to delay distortions exceeding 50 ms alter the short time amplitude spectrum and are thus audible [Schroeder, 1966].

However, experiments using sinusoidal tones have demonstrated that the ear is sensitive to the signal waveform - ie the phase relationship of the tones. Different waveforms with identical amplitude spectra but differing phase relationships are perceived as sounding different [Mathes et al, 1947; Plomp et al, 1969].

Using the model of Fig. 4.5, the early channel vocoders gave synthesised speech that sounded mechanical and was often described as "mushy" because the harmonics of the local voiced excitation source were all of identical (zero) phase. Experiments to include speech-like phase [Gold, 1964] in the voiced segments gave far more natural sounding speech. In addition, careful attention to the design of the analysis/synthesis filters to equalise the amplitude response and group-delay across all the filters gave a marked improvement in vocoder performance - implying better re-synthesised speech [Golden, 1968]. For the more recent Linear Prediction (vo)Coders (LPC), adding

a random element to the phase angles of the excitation source gives useful improvement in the synthesised speech quality [Kang et al, 1983].

Conversely, in situations that require enhancement of the intelligibility of the speech (eg aircraft cockpit communications), particularly noisy speech [Lim et al, 1979], it has been shown that the Fourier phases of the speech signal are unimportant [Wang et al, 1982].

Based on an experiment conducted in 1961, this chapter describes manipulation of speech phase by a different method and attempts to answer some important questions raised by this early experiment.

5.2 DAVID'S EXPERIMENT AND QUESTIONS RAISED

In brief, David et al [1961] manually extracted the pitch periods of male and female voiced speech. These were then Fourier analysed to extract the Fourier coefficients

$$c_k = A_k + j\,B_k \quad , \qquad k = 1,\ldots M$$

$$= |c_k| \mathrm{Arg}(c_k) \tag{5.1}$$

The speech was then re-synthesised via the Fourier series

$$f(n) = \frac{A'_0}{2} + \sum_{k=1}^{M-1} A'_k \cos kn + \sum_{k=1}^{M-1} B'_k \sin kn \tag{5.2}$$

$$\text{with } A'_0 = A_0$$

where the desired phase patterns are established in 3 ways:

(a) cosine re-synthesis:

$$A'_k = \sqrt{(A_k^2 + B_k^2)}, \quad k = 1,\ldots M$$

$$B'_k = 0$$

(b) sine re-synthesis:

$$A'_k = 0$$

$$B'_k = \sqrt{(A_k^2 + B_k^2)}, \quad k = 1,\ldots M$$

(c) random sign inversions of the coefficients A'_k of the cosine re-synthesis with 4 different sign combinations.

They found the random sign inversions gave a speech envelope (ie the instantaneous envelope m(t)) with a lower peak factor than the original, while the alternatives of (a) and (b) resulted in higher peak factors. In addition, the higher peak factor speech sounded "raucous", while the lower peak factor sounded more "tonal" (smoother?), but that this effect appeared to be highly variable. They reported that naive listeners showed only slight ability on average to distinguish high and low peak factors, but that expert listeners were considerably more successful. The authors concluded that phase information was important to speech quality, particularly when listening over headphones.

5.2.1 Questions Raised by the Experiment

The Fourier analysis/synthesis technique with manual pitch period detection is cumbersome. To automate the procedure would require accurate voiced-unvoiced decisions [Siegel, 1979; 1982] coupled with accurate delineation of individual pitch periods [Rabiner et al, 1976] for use with a DFT. Such procedures exist and can be automated, but are themselves computationally intensive and not necessarily with the required level of accuracy across all speakers.

Thus:

Question 1 : Is there a simpler method that can be used instead of the Fourier analysis/synthesis approach?

With regard to the three approaches for re-synthesising the signal, it is not immediately obvious why the random sign inversion approach is the only approach that gave a peak factor reduction. Furthermore, the amount of reduction achieved is not indicated. In the context of telephone bandwidth speech and communications systems, it is known that the peak-to-rms-ratio of speech is large (upto 20 dB [Ahlborg,1978]. This measure is an alternative to David's peak factor. Purton [1962] also indicated that a PCM system should be able to support a signal dynamic range of at least 38 dB.

Traditional measures employed to constrain the dynamic range of speech are logarithmic companding (in PCM systems) and clipping. Both of these processes are highly non-linear. In addition, while it is very easy to implement clipping, such a process introduces spectral distortion and is non-reversible.

The implication of the comments above are that a relatively

simple, reversible technique that helps to limit dynamic range of speech may be worthwhile. Thus:

Question 2 : What form of phase manipulation is optimal in giving the lowest peak factor, and is it achievable?

Question 3 : How much reduction in peak factor is available?

Question 4 : What, if any, is the effect on the pdf of the speech signal?

Finally, the original experiment reported using male and female speech, but no indication is given of any differences in behaviour. Thus:

Question 5 : What, if any, are the differences in the behaviour of male and female speech that is subjected to phase manipulation?

5.2.2 Towards Some Answers

Given the early work of previous chapters, an answer to Question 1 has already been indicated. In fact, a non-linear phase FIR filter is used to manipulate the phases of the speech signal. This approach takes advantage of the knowledge of FIR filters and techniques (digital signal processing using computers) that were not available at the time of the original experiment. Since the procedure of filtering is linear, such an operation is reversible provided exact mirror images of the phase responses can be generated. If a design method exists, and is available for non-linear phase filters, then producing mirror image phase responses in two filters is trivial. Thus the process of phase manipulation by filters is easily and exactly reversible provided there are no other group-delay distortions (phase shifts) introduced in the signal by the intervening channel.

However, the larger question of what form of phase response is as yet unanswered. Furthermore, how does the process of linear filtering using a FIR filter compare with the original method of Fourier analysis/synthesis? These, and the other questions are pursued in the sections that follow.

5.3 SPEECH DATA BASE AND EXPERIMENTAL ENVIRONMENT

5.3.1 Digitising the Speech

All the work to be described is based on using real speech of bandwidth 0-3400 Hz that has been digitised and stored on a DEC VAX

11/750 computer. An identical 5 second phrase spoken by 2 adult males and 2 adult females in English forms the principle data base. These are identified as M1, M2 and F1, F2. The method used to establish the data base is shown in Fig. 5.1.

The speech is recorded via an electret microphone on a high quality cassette tape recorder using Dolby C noise reduction and whose minimum S/N ratio specification on replay is 68 dB. This is then replayed through an anti-aliasing filter and digitised by a 12-bit linear A/D converter (DEC LPA-11K Subsytem with AD-11K A/D and AA-11K D/A) at a sampling rate of 8064 Hz (the nearest the CPU clock could get to 8000 Hz) and stored on disk. The speech is DC isolated from the A/D converter to avoid offsets, and the buffer amplifier lifts the analog signal to occupy as much of the ±5v dynamic range of the A/D converter as possible. This serves to maximise the signal-to-quantisation noise ratio.

The speech from male M1 and female F1 was recorded in an anechoic chamber, while that for M2 and F2 was recorded in a carpeted domestic living room at night time with no traffic noise. Thus, the residual noise level of the room to the degradation of the apparent S/N ratio of the speech is ignored.

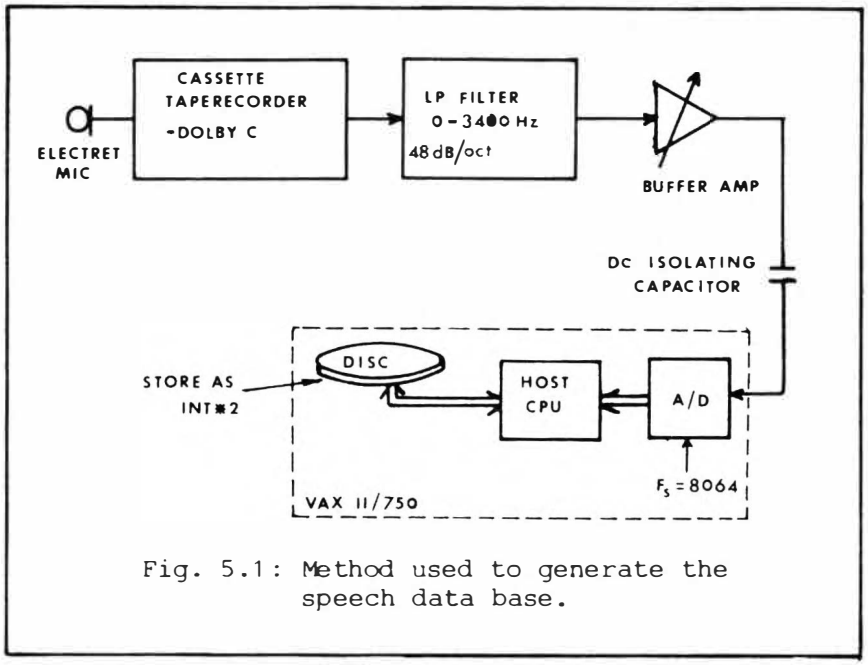


Fig. 5.1: Method used to generate the speech data base.

5.3.2 Principles of the Computer Simulations

The principal aspects related to the computer simulations are shown in Fig. 5.2.

All calculations and analysis are done in single precision floating point arithmetic. Thus the digitised speech from the A/D, although stored as 2-Byte integers, is first converted to floating point representation before computation. The results of the simulations are stored on disk as Floating point numbers (REAL*4) and are only truncated back to 2-Byte integers when the speech is sent to the D/A converter (12-Bit) for conversion back to an analog signal.

When filtering of the data is performed, the software implements the direct form of the convolution relation. ie, the relation

$$y(n) = \sum_{k=0}^{N-1} x(n)h(n-k) \qquad (5.5)$$

is implemented directly. This is to avoid any problems or arguments related to the use of faster methods such as the FFT based overlap-add method [Helms, 1967].

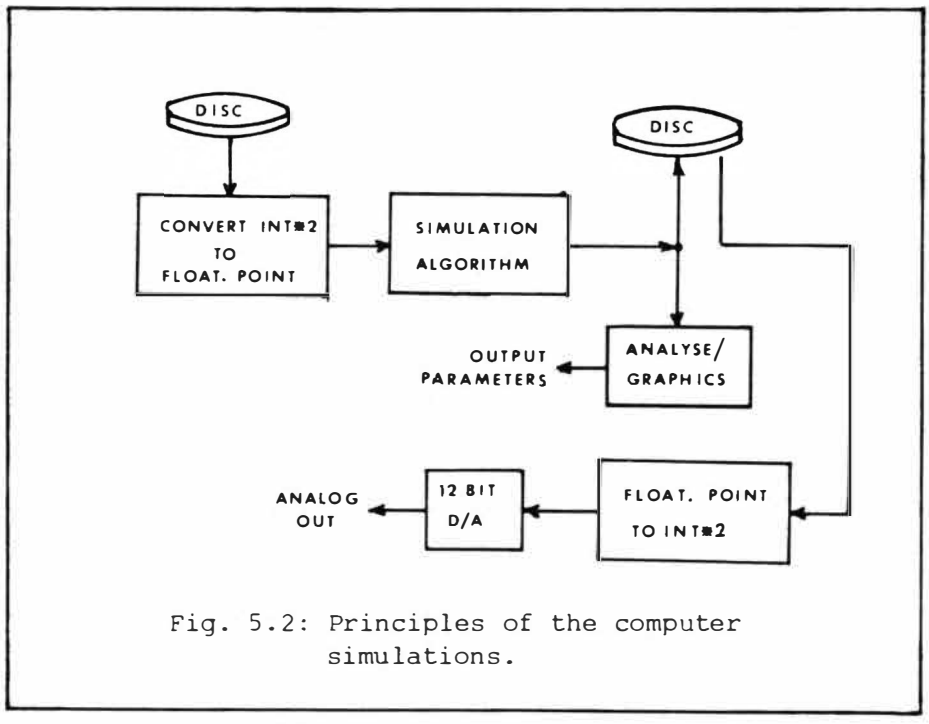


Fig. 5.2: Principles of the computer simulations.

To compute the analytic envelope, the Hilbert transform $\hat{s}(t)$ of $s(t)$ is required. The $\hat{s}(t)$ is generated by using a Hilbert FIR filter designed by the computer program developed in Chapter 2. The actual computer implementation of equation (5.5) is listed in Fig. 5.3 and

```
C-------------------------------------------------------------
C                SUBROUTINE FILTER
C
C  Implements the convolution equation directly. Note that
C  because the filter may be  non-linear phase, NPOINTS is
C  the total filter length, and FILTERMS holds the complete
C  impulse response.
C
C     NPOINTS - the number of filter terms
C
C     SIZE    - the length of the speech buffer arrays
C
C     FILTERMS - the array holding the filter terms
C
C     INPUT    - the input speech data
C
C     OUTPUT   - the filtered speech
C-------------------------------------------------------------
       SUBROUTINE FILTER(NPOINTS,SIZE,FILTERMS,INPUT,OUTPUT)

       INTEGER NPOINTS, SIZE
       REAL    FILTERMS(NPOINTS), INPUT(SIZE), OUTPUT(SIZE),
     +         SUM

C
C  to elminate the inherent time delay and allow time
C  matching of signals, start with the half the window into
C  the data
C
       L=NPOINTS/2
       IF(MOD(NPOINTS,2).GT.0) L=L+1  !if filter has odd length
       DO I=1,SIZE
         SUM=0.0
         DO K=0,NPOINTS-1
           J=L-K
           SAMPLE=0.0
           IF(J.GE.1 .AND. J.LE.SIZE) SAMPLE=INPUT(J)
           SUM=SUM+SAMPLE*FILTERMS(K+1)
         END DO
         OUTPUT(I)=SUM
         L=L+1
       END DO
       RETURN
       END
```

Fig. 5.3: VAX FORTRAN code that implements the convolution relation of equation 5.5. THe subroutine ensures time synchronicity (sample index n) between input and output.

eliminates the inherent (N-1)/2 (sample) time delay, where N is the filter length. This guarantees time synchronicity (sample index n) between s(t), $\hat{s}(t)$ and the envelope m(t).

The software uses dynamic array storage (LIB$GET_VM) for the data, so for simplicity, both INPUT and OUTPUT arrays are of identical length. The program internally rejects the first and last (N-1)/2 signal samples corresponding to the edge effects of the convolution process.

5.3.3 Establishing the Signal Extrema

In section 4.3.2, two attempts by Brady to overcome the difficulties of measuring the "real" peak values of speech were mentioned. However, one important criterion for these methods is that at least two seconds of speech were required to produce a realistic result. This makes the methods unsuitable in situations where single words that occupy less than a second of elapsed time are the subject of analysis.

Detecting the instantaneous sample maximum value in some interval is recognised as an unrealistic measure in an actual (telecommunications) environment because of the likelihood of extraneous noise spikes which would give excessively large values.

However, from section 5.3.1, all the speech used in this work has a high S/N ratio and can be considered as "high quality". Using the cassette recorder S/N ratio specification, 68 dB below 5V is 2mV which is less than ±1 count of the A/D converter (±1 count = 2.44 mV re 5V). Thus, for this work, the arguments above concerning the real or effective signal extrema are ignored, and the signal extrema s_{min}, s_{max} are defined as

$$s_{max} = \max[s(n)], \quad s(n) \geq 0, \quad n = 1,\ldots,M$$
$$s_{min} = \min[s(n)], \quad s(n) < 0 \qquad (5.6)$$

where M is the number of signal samples searched. No threshold is used to exclude very low signal sample values that might be considered as noise.

5.3.4 Peak Factor Defined

In the original experiment, David et al used change in peak factor of the envelope m(t) to characterise the differences in phase patterns. Since the real interest is on the effect on the speech signal itself, the measurements reported here are made on the signal s(t) rather than the envelope m(t). m(t) is used here mainly as a descriptive tool that show much more clearly the effect of phase manipulation on a complex signal composed of many harmonics.

(i) segmental peak factor PF_{seg}

$$PF_{seg}(j) = \frac{s_{max} - s_{min}}{\frac{1}{M}\sum_{k=jP+1}^{jP+M}|s(k)|}, \quad j=1,\ldots Q \tag{5.7}$$

This segmental approach implicitly recognises the non-stationary nature of speech. M is the short time length selected, in samples; P is the interval, in samples, by which the short time segment is stepped and j is the j th segment. Usually P=M/2 will be used. Hence, the number of segments Q=(L-M+1)/P. For example, if the segment length required is Δt=20 ms with a sampling rate of 8064 Hz, then M=Δt*Fs=161 and P=80.

(ii) Averaged Peak Factor PF_{av}

The segmental description is useful in demonstrating the dynamic behaviour of the peak factor for short segments or single words, but for long periods of conversational speech it is inappropriate. This is simply because the number of segments would be so large that a plot of PF_{seg} would be so compressed that it would be difficult to observe differences between the original and phase randomised speech. In this case, PF_{av} is used as the represenative measure.

$$PF_{av} = \frac{1}{Q}\sum_{j=1}^{Q} PF_{seg}(j) \tag{5.8}$$

5.4 AMPLITUDE AND ZERO STRUCTURE OF VOICED SPEECH

5.4.1 A Synthetic Signal

For clarity and simplicity, the discussion begins by analysing a synthetic signal of three harmonics constructed from a sine series:

$$f(n) = \sum_{k=1}^{3} A_k \sin\left(\frac{2\pi f(k) n}{F_s} + \theta_k\right) \quad , n=1,... \tag{5.9}$$

where the θ_k are the starting phases. The example is restricted to 3 harmonics because the equivalent algebraic polynomial is a quadratic which is easily handled analytically. Building on the discussion of section 4.6.2, the purpose of this section is to demonstrate, by example, the effect on the magnitudes of the signal and its envelope, of modification of the analytic zero positions.

Using

$$\begin{aligned} A_1 &= 0.6 \\ A_2 &= 1.0 \\ A_3 &= 0.8 \\ \theta_k &= 0.0, \quad k = 1,..,3 \\ f_1 &= 100 \text{ Hz} \end{aligned} \tag{5.10}$$

Fig. 5.4 shows one period of f(t) of equation (5.9), its analytic envelope and its analytic zeros $\{z_1, z_2\}$. z_1 and z_2 are obtained according to the following: since f(t) is composed of a sine series

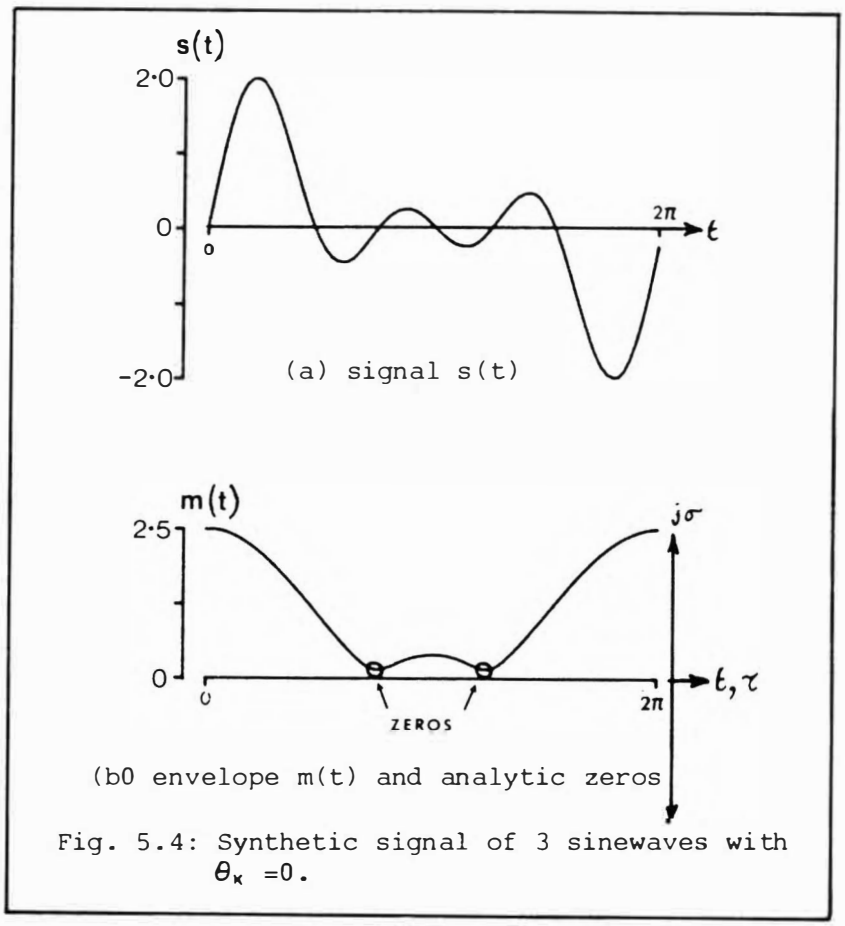


Fig. 5.4: Synthetic signal of 3 sinewaves with $\theta_k = 0$.

(odd symmetric sequence), then its Fourier Transform $F(\omega)$ is pure imaginary with $Re\{F(\omega)\} = 0$. Hence, its Fourier coefficients c_k are given by

$$|c_k| = A_k$$
$$Arg(c_k) = \tan^{-1}\{Im\{F(\omega)\}/0\} = \pm\pi/2 \qquad (5.11)$$

Inserting the single period of f(t) of Fig. 5.4 into MXFFT, and recalling the discussion in Chapter 1.0 on the practical aspects of using an FFT, the single cycle of Fig. 5.4 appears with negative sign components in the FFT. Thus, in this case, $Arg(c_k) = -\pi/2$, $k=1,\ldots,3$. Hence, using these c_k in equation (4.17):

$$\sum_{k=1}^{3} c_k x^k = -j0.6x - j1.0x^2 - j0.8x^3 = 0 \qquad (5.12)$$

which, on normalising gives

$$x^2 + 1.6667x + 1.3333 = 0 \qquad (5.13)$$

and has solution

$$x = -0.8333 \mp j0.7993 \qquad (5.14)$$

Computing the analytic zeros $\{z_1, z_2\}$ from equation (4.16) (and ignoring the $1/\Omega$ terms) gives

$$\tau_1, \tau_2 = Arg(x) = 2.377, -2.377$$
$$\sigma_1 = \sigma_2 = -\ln|x| = -0.144 \qquad (5.15)$$

and are plotted on the same axes as the envelope of Fig. 5.4. The solution to the above problem via the algorithm CPOLY according to section 4.6.3 and subsequently equation (4.16) (again neglecting the $1/\Omega$ terms) is

$\tau_1 = 2.3690$ $\qquad$ $\tau_2 = -2.3974$
$\sigma_1 = 0.1291$ $\qquad$ $\sigma_2 = 0.1280$

The minor differences between these results and the analytic solution above is mainly due to the result of the FFT routine which computed the c_k as

$c_1 = 0.608\ \underline{/-1.5587}$
$c_2 = 1.00\ \underline{/-1.5440}$
$c_3 = 0.785\ \underline{/-1.5303}$
where /___ is the phase angle.

ie - the FFT computed phases are slightly in error.

Voelker has demonstrated that the envelope fluctuations are invariant under zero conjugation - ie by substituting $-\sigma$ for $+\sigma$ (or vice versa). Observe that the zeros in Fig. 5.4 are bunched in time and in chapter 4.0 it was observed that regions of low envelope amplitude (the dips) corresponded to the τ_k with the intensity of the dips governed by $|\sigma|$. Holding σ constant and moving the zeros apart in time so that they are evenly spaced (with reference to the centre of the interval) gives τ_1', τ_2' = 1.5707,-1.5707. Using these new τ'_k and σ and working backwards through the calculations (through equation (4.16) and then equation (4.18)) generates the equivalent polynomial to equation (5.13) as

$$f'(z) = 0.6x^2 + 0.8 = 0 \qquad (5.16)$$

This new polynomial shows that only two of the original harmonics A_1 and A_3 are required which means that f'(t) is a completely different signal to f(t) in terms of information content and signal power because the harmonic A_2 of the original signal has been completely eliminated. Thus, comparison of peak factors and envelope structure between the two signals is pointless. Similarly, holding the τ_1, τ_2 constant and altering $|\sigma|$ alone can be demonstrated to alter the equivalent polynomial in both magnitude and phase and therefore gives yet another completely different signal.

The discussion above is a long winded way of saying there are an infinite number of signals f(t) of equation (5.13) that possess two analytic zeros per period. For a speech signal, it has already been indicated that it is the amplitude spectrum that dominates speech intelligibility. Thus, modification of $|c_k|$ is prohibited. With this constraint, the only parameter left to manipulate are the Fourier phases which will give rise to simultaneous changes in each τ_k and σ_k.

The principal requirement therefore, is to find one or more (starting) phases θ_k of equation (5.9) that will produce a set of zeros $\{z_1, z_2\}$ such that τ_1, τ_2 are evenly distributed in the period and the $|\sigma_1|$, $|\sigma_2|$ are sufficently large to minimize the dips in the envelope waveform. Clearly, from equation (5.9), there are many possible signals with different combinations of θ_k. Of this set, one particular combination will give a signal with the lowest possible peak factor.

Even for this example, establishing the required Fourier series coefficients c_k and thus θ_k of equation (5.9) from the required zero positions is not a trivial problem. Appendix 5.0 details the procedure involved merely to set the positions of τ_1, τ_2. One solution suggests that $\theta_1 = 2.435$ with θ_2, θ_3 unchanged. Bearing in mind the discrepancies between the FFT generated coefficients and the analytic solution demonstrated above, using $\theta_1=\pi$ in the FFT generated solution gives the results shown in Fig 5.5 where

$\tau_1 = -0.01211$ $\quad\quad$ $\tau_2 = 3.1283$
$\sigma_1 = 0.8303$ $\quad\quad$ $\sigma_2 = -0.5396$

Notice that $|\tau_2 - \tau_1| = 3.1404$ ie the zeros are virtually equidistant from each other. This is just one particular solution that gives an equidistant spacing for the τ_k. Whether this solution corresponds to the unique solution that gives the lowest possible peak factor is not apparent.

An additional important result of Appendix 5.0 is that for the analytic zeros $\{z_k\}$ to be evenly distributed in time, the zeros $\{x_k\}$ of the signal must be entirely real. This implies that the signal

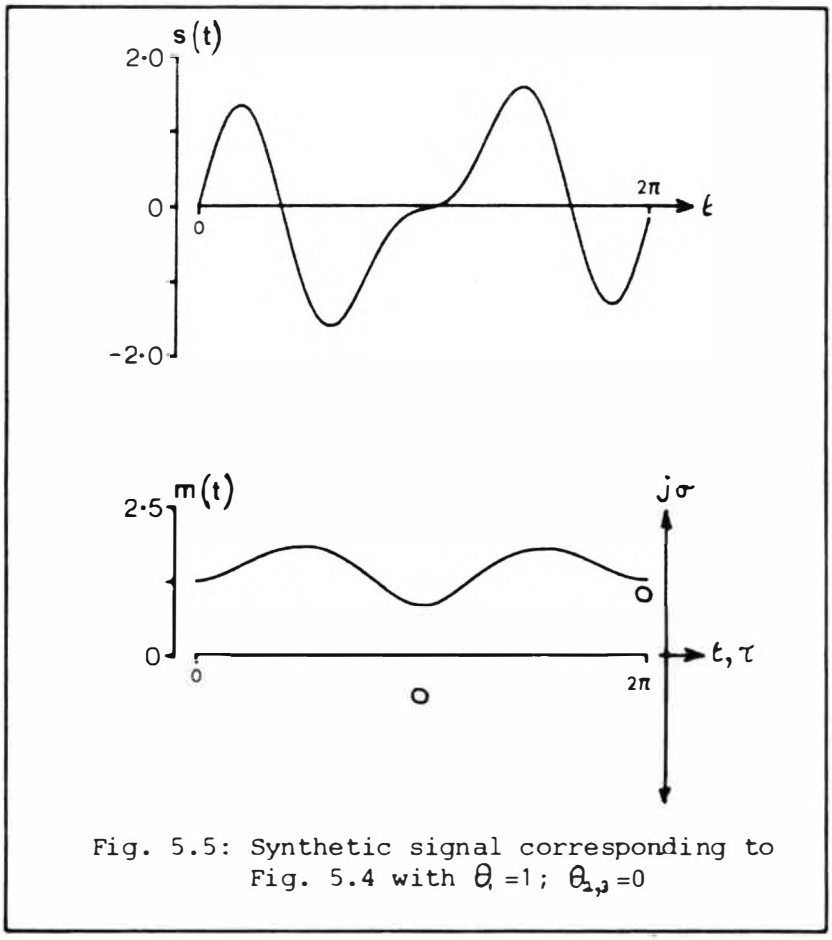


Fig. 5.5: Synthetic signal corresponding to Fig. 5.4 with $\theta_i = 1$; $\theta_{j,k} = 0$

possess only real zero crossings. Discussion of this point is deferred to section 5.6.1 when a real word is examined.

For Fig. 5.4, the peak factor according to equation (5.8) is 5.78 and that for Fig. 5.5 is 3.85. Since the amplitude spectrum is identical for both signals, the power as given by the long term rms level must be identical. Hence the drop in peak factor must represent a change in the signal dynamic range ($s_{max} - s_{min}$) of $20\log_{10}(5.78/3.85)$ = -3.52 dB

The foregoing discussion demonstrates that direct zero manipulation is cumbersome - even for the simple example. Manipulation of the Fourier phases $Arg(c_k)$ is relatively easy, but establishing the required phase values to give the lowest peak factor is an extremely difficult problem. The principal requirement of the Fourier phases is to alter the positions of the zeros so that they are evenly distributed in time.

5.4.2 The irreducible minimum Peak Factor

The arguments above indicate that finding the minimum peak factor possible for a real signal may be an unsolvable problem. However, the irreducible minimum peak factor that any signal may have is proposed here.

One of the requirements of the zero positions is that $|\sigma|$ be as "large" as possible to minimise the intensity of the dips in the envelope. The case where the envelope, in the limit, smooths out to a straight line (ie a constant) corresponds to a signal composed of a single sinusoid: equation (4.4) gives $\sqrt{(\sin^2+\cos^2)}=1$. It is well known that the peak and average values of a sinusoid are related by

$$\frac{2}{\pi} s_{max} = s_{av} \qquad (5.17)$$

where s_{av} is the absolute average value. For a sinusoid, $|s_{max}|=|s_{min}|$ so equation (5.17) is simply a rearrangement of equation (5.7). Hence

$$PF_{seg} = 2s_{max}/s_{av} = \pi$$

Thus a single sinusoid has a peak factor of π. Furthermore, since a sinusoid has constant periodicity, PF_{seg} over one cycle is the same as PF_{av}. This value of peak factor must be the limiting value for all signals, because a real signal is composed of more than one sinusoid - ie, it has more than one analytic zero. Under these conditions, a real

signal cannot have an envelope that is a straight line. Exactly how close to this minimum a real signal can get remains dependent on being able to establish the unique set of zeros.

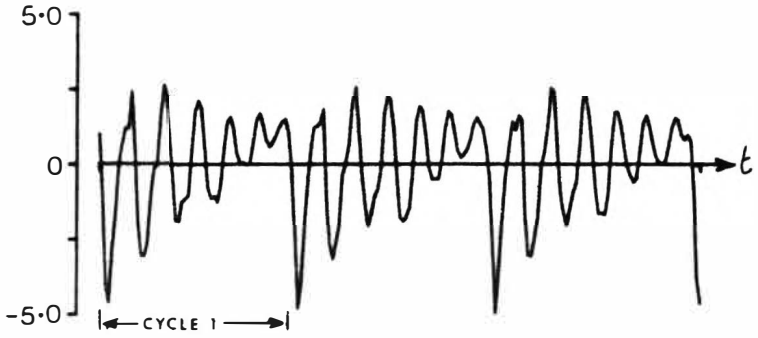
On this basis, comparing Fig 5.4(a) with Fig. 5.5(a) shows that spacing the analytic zeros evenly in time has resulted in a signal whose time behaviour is not far removed from that of a single sinusoid and has a peak factor of 3.85.

5.4.3 Real Speech Segment

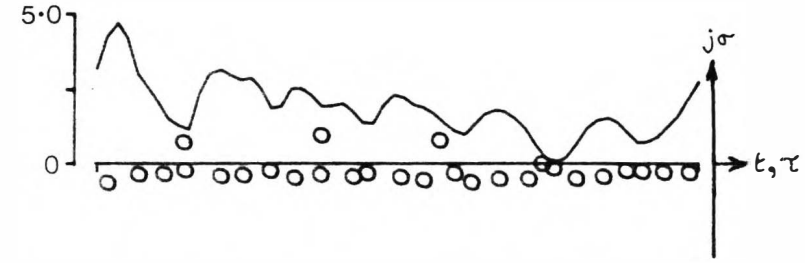
Fig 5.6(a) shows 3 cycles of the full bandwidth vowel (0-3400 Hz) /a/ from the word "fast" spoken by male M1. Using "cycle 1" shown , Fig 5.6(d) is the amplitude spectrum of this cycle and Fig 5.6(b) its analytic envelope. The FFT length (MXFFT) used was 60, corresponding to the number of signal samples available in cycle 1. There are thus only 29 positive harmonics which therefore indicates the envelope has 28 analytic zeros and these are shown in Fig. 5.6(b). Fig 5.6(c) is the envelope and zeros corresponding to the first 6 harmonics and shows that it is these few harmonics (zeros) that form the basic shape of the envelope (and thus the signal). Observe that the zeros in Fig. 5.6(c) are not evenly distributed along the real time axis. That these few harmonics are the important ones is not surprising since, in Fig. 4.8 it was shown that most of the speech energy is concentrated in the frequency range 100-800 Hz which corresponds to the first formant.

The remaining 23 zeros of Fig. 5.6(b) are evenly distributed close to the real time axis and correspond to the bandwidth above the 6th harmonic. Their principle effect on the basic envelope shape is to add the "fine structure".

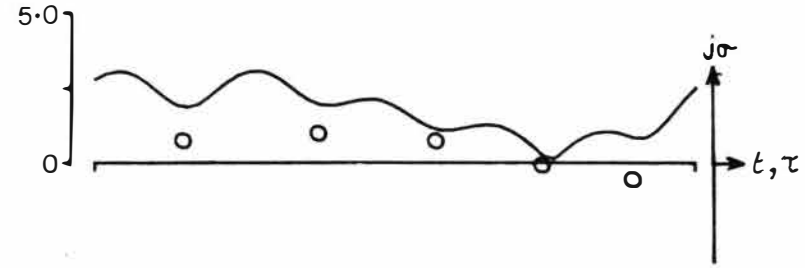
Careful observation of the 2nd and 3rd cycles of Fig 5.6(a) reveals subtle differences in the signal waveform. These differences will be manifested as small changes in the zero positions. Thus it is apparent that even within the usually accepted "short-time" interval in which voiced speech is considered stationary, there are minor shifts of the zero locations about some mean position. These localised zero drifts are a manifestation of the observation in chapter 4.0 that the harmonic structure of voiced speech is not constant and is more accurately represented as "teeth" - Fig. 4.10.



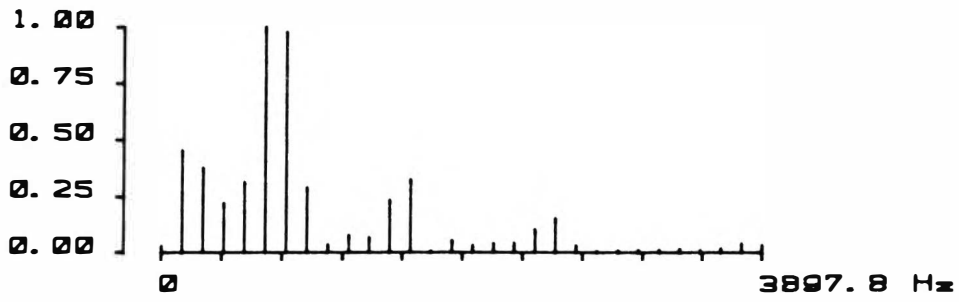
(a) 3 cycles of the signal s(t)



(b) envelope and zeros of the full bandwidth "cycle 1".



(c) envelope and zeros of a signal composed of the first 6 spectral lines of part (d)



(d) amplitude spectrum of "cycle 1" of part (a)

Fig. 5.6: Envelope, zero structure and amplitude spectrum of a single cycle of the vowel /a/.

5.4.4 Summary

The discussion of section 5.4 demonstrates the following key points:

1. Intervals of low zero density correspond to high amplitude envelope (and thus signal) amplitudes. Hence
2. The important zeros that determine the basic envelope shape must be evenly distributed in time.
3. One unique set of zeros (evenly distributed) will give a signal with the lowest possible peak factor.
4. Establishing the unique set of zeros that gives lowest peak factor signal is in practice an extremely complex problem. As far as this author is aware, no such solution to finding the unique zero set has been found.
5. The important zero positions of (voiced) speech are not constant but suffer localised drift in a region. This is attributable to the well known time varying harmonic structure of speech. The implication of this is that the unique set of zeros that gives the lowest peak factor signal changes with every pitch period.
6. Given 1. and considered from the Fourier domain, locations of high signal amplitudes correspond to instances where the Fourier phases of the signal components all add constructively.
7. The irreducible minimum peak factor of a signal is π. This corresponds to the value achieved by a single sinusoid. Since a real signal contains many sinusiods, this minimum will never be reached in practice.

The conclusion from the above points is that manipulation of the Fourier phases by some arbitrary randomisation process is likely to lead to a signal with a lower peak factor than the original. However, such a process is unlikely to give the lowest possible peak factor.

In constrast, David's method (c) of section 5.2.1 is equivalent to using equation (5.9) with a cosine synthesis and $A_k = A'_k$ of equation (5.4) and the θ_k either 0 or π.

5.5 REQUIREMENTS OF THE NON-LINEAR PHASE FILTERS

The previous section established that the only practically useful way of manipulating the zeros was by some procedure to assign, or modify, the Fourier phases of the signal. A non-linear phase FIR

filter is a simple, effective method of providing arbitrary phase manipulation of the speech signal. The principal reason for choosing FIR filters is that they are guaranteed to be stable and it is, at least initially, easier to generate the required impulse response coefficients.

There are however, two principal problems in generating the required filter impulse response. The first is related to the basic properties of voiced speech and defining the required non-linear phase characteristic. The second problem is a design problem once the phase response has been defined.

The first problem has 3 aspects:

(1a) the actual Fourier phase values of the speech will vary continuously between different sounds for the same speaker, as well as between speakers.

(1b) the fundamental frequency also varies continuously.

(1c) the fundamental frequency and therefore the frequency spacing, of female voices is in general twice that of males - Fig. 4.11.

Since it is not possible to establish exactly what phase values for each harmonic are ideal (in giving the lowest peak factor), the assumption is made that large phase differences between adjacent harmonics are required. The three 3 aspects above suggest a phase response shown in Fig 5.7(a).

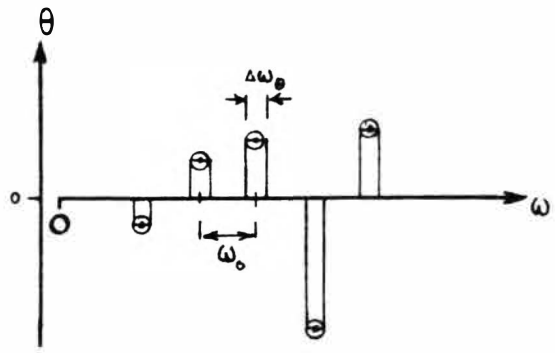
A uniform distribution random number generator is used as the basis of generating numbers in the range $(-\pi, \pi)$ which are used as the phase values θ_k . These θ_k are centered on the average harmonic spacing ω_o indicated by the $(\odot)$, and having a width $\Delta\omega_\theta$ which is the "jitter" of (1b) above (the "teeth" of Fig. 4.10). For males, the average fundamental frequency is around 100Hz and for females around 200Hz. Thus the narrower harmonic spacing of the male voice is chosen.

Given this phase specification, the design problem has three principle aspects:

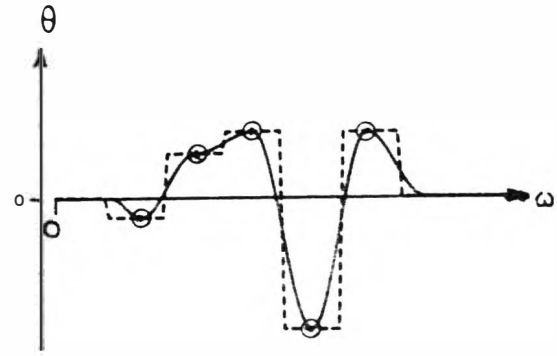
2a. A flat (minimum ripple) passband is required to avoid distortion of the amplitude spectra. As a worst case, the maximum allowable passband ripple for this work is set at ± 2 dB.

2b. The phase discontinuities of Fig. 5.7(a) are discontinuities in the real and quadrature parts of the frequency response of the filter.

2c. It is well known that truncating (limiting) the impulse response



(a) random phase values θ centred on a mean harmonic spacing ω_o and phase width $\Delta\omega_\theta$



(b) smoothed version of the ideal in (a)

Fig. 5.7: Specifying the non-linear phase response of the filters.

(time) length of a sequence with such discontinuities introduces ringing (Gibbs phenomenum). Such ringing is minimised by increasing the length of the filter. This implies that very long filter lengths will be required.

The discontinuities of Fig. 5.7(a) are minimised by widening the very narrow interval $\Delta\omega_\theta$ to be the same as the average spectral interval($\odot$). Additionally, the phase response is smoothed to eliminate the discontinuities by interpolating smoothly between the ($\odot$) and ensuring the phase response goes smoothly to zero at the passband edges. This procedure gives the phase response of Fig. 5.7(b).

5.5.1 Parameters of the filters used for the simulations

Given the arguments above, the filters used for the simulations to come use $\Delta\omega_\theta$ = 100 Hz, which are centred on the mean harmonic

positions corresponding to ω_0=100 Hz.

Section 5.3 above suggested that the number of important zeros (phases) that require manipulation are few and correspond broadly to the harmonics lying in the first formant. This suggests a non-linear phase response in the interval 0-1000 Hz. However, as a check on this assumption, two extra identical sets of filters with a phase bandwidth of 0-2000 and 0-3000 Hz are also used. In the text that follows, the term "phase bandwidth" is used to refer to the frequency interval over which a non-linear phase is defined.

When describing the design of non-linear phase filters in section 3.6, equation (3.20) indicated two ways to assign the random phase values θ_k. These are referred to here as r_1 corresponding to equation (3.20a) and r_2 corresponding to equation (3.20b). Thus, with the three phase bandwidths indicated above, there are six different filters. The phase responses of all these filters are generated from a single original random sequence. Thus the phase bandwidth of 0-2000 Hz duplicates the phase response for the interval 0-1000 Hz and then uses more of the random sequence to continue out to 2000 Hz. Similarly with the filters with phase bandwidths of 0-3000 Hz.

These filters will be referred to as FILT1-FILT6. Their essential parameters are shown in Table 5.1 and were designed using ALgazi's algorithm.

The choice of filter length N=299 results from the conflicting requirements of trying to minimise the length N, keep the passband ripple $\delta_p \leq 1$ dB and the stopband ripple $\delta_s \leq 50$ dB. Fig. 5.8 shows the phase response and passband response of FILT5. It is apparent that the passband has significant ripple in the region 2300 - 2500 Hz approximately. Since this band is well up into the 3rd formant and contains little signal energy (Fig. 4.8), its effects on the results of the simulations are considered to be insignificant.

5.6 PHASE RANDOMISATION OF A SINGLE WORD

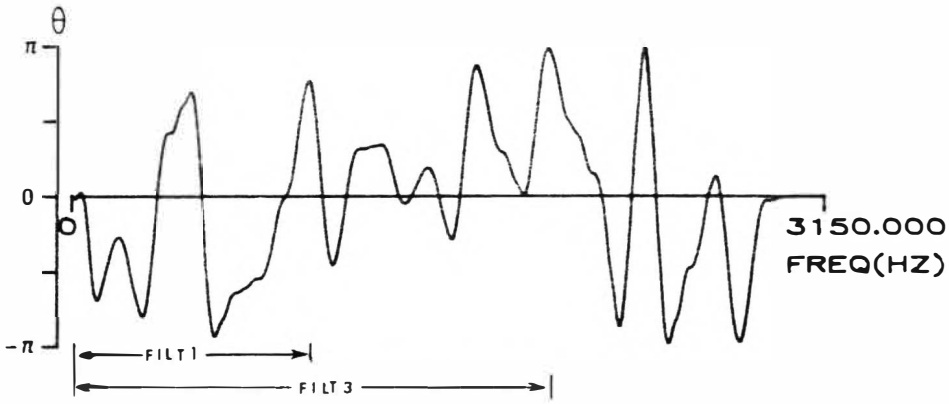
The word "fast" spoken by male M1 is used here to demonstrate the effects of phase randomisation. This section compares David's method (c) of section 5.2.1 with randomisation achieved using the six filters described above.

Neiderjohn et al [1984] investigated the relation

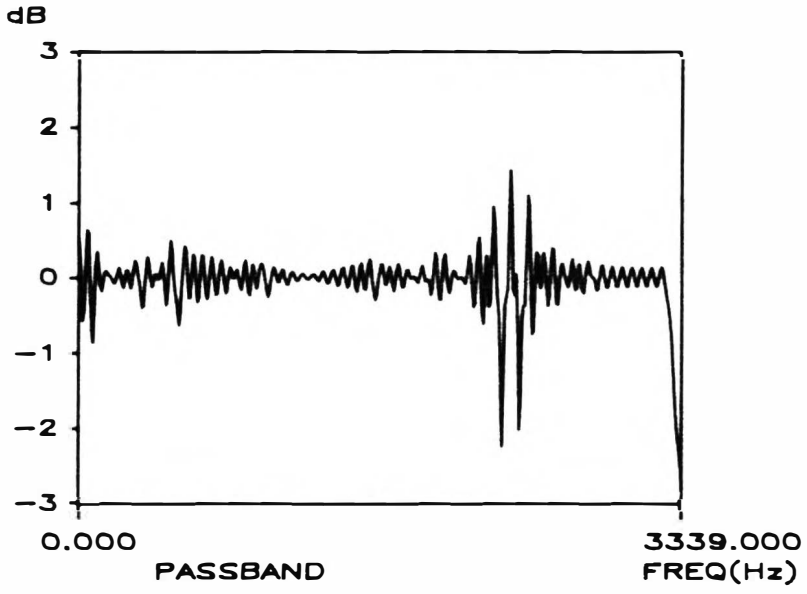
TABLE 5.1: ESSENTIAL PARAMETERS OF THE SIX FILTERS USED IN SECTIONS 5.6 AND 5.7.1

Power Gain = 1.0
N= 299, Iterations=25, A = 10^4
$\Delta\omega_\theta$ = 100 Hz, ω_o = 100 Hz

	phase bandwidth					
	0-1000 Hz		0 -2000 Hz		0 - 3000 Hz	
NAME ->	FILT1	FILT2	FILT3	FILT4	FILT5	FILT6
SEQUENCE	r_1	r_2	r_1	r_2	r_1	r_2
RIPPLE: δ_p (dB)	-0.86	-0.79	-0.87	-1.81	-2.24	-1.07
δ_s (dB)	-57.5	-59.0	-58.6	-57.8	-56.3	-61.3



(a) phase response showing how the phase responses of FILT1 and FILT3 are contained as a subset.



(b) passband response

Fig. 5.8: Passband and phase response for FILT5. This filter has the largest passband ripple.

$$\beta = \frac{s_{av}}{s_{rms}} \tag{5.18}$$

where

$$s_{av} = \frac{1}{L} \sum_{k=1}^{L} |s(k)| \tag{5.19}$$

$$s_{rms} = \left(\frac{1}{L} \sum_{K=1}^{L} s^2(k)\right)^{\frac{1}{2}} \tag{5.20}$$

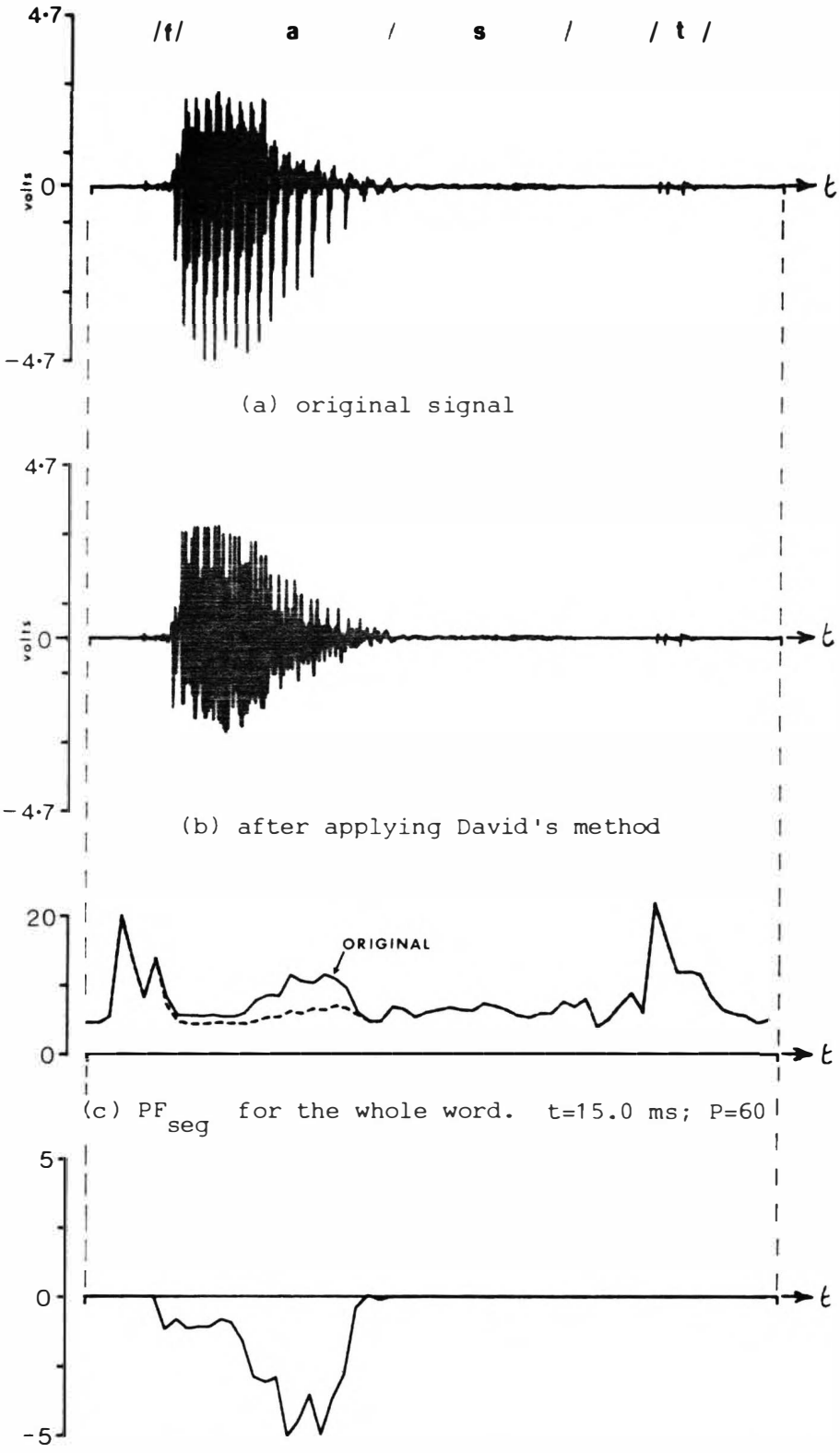
and L is the total number of signal samples available. Their results are shown in Fig. 4.7. The point here is that the two different measures of signal power of equations (5.19) and (5.20) give different results. The random phase filters that will be used in sections 5.6 and 5.7 to come are designed to have unity power gain. This attribute of the filters gives identical values of s_{rms} for the original and its phase randomised version, but not s_{av}. Since the denominator of PF_{seg} in equation (5.7) is a segmental version of equation (5.19), all the phase randomised signals are scaled so as to make s_{av} identical to the original signal.

Using David's method, Fig. 5.9 shows the original signal, the result from the analysis/synthesis over the region of the vowel /a/ and plots of PF_{seg} of each signal. Observe there is consistent reduction in peak factor over the segment /a/.

Figs. 5.10,5.11 and 5.12 are the PF_{seg} plots for the six filters. Part (a) of each figure corresponds to the sequence r_1, and part (b) corresponds to the sequence r_2 (equation (3.20a) and (3.20b)). Part (c) of each figure represents the difference (or change) in peak factor corresponding to the difference between the two traces of each of the (a) and (b) parts of each figure.

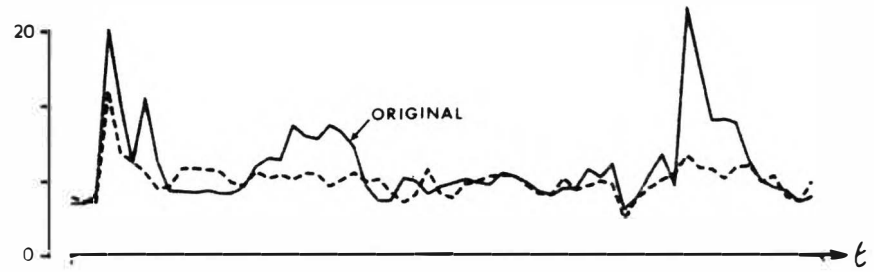
The important points of these results are

1. the peak factors of the signals filtered by FILT1, FILT3, FILT5 corresponding to the sequence r_1 all show regions of the voiced segment /a/ were the peak factor is increased (the (a) parts of the figures)
2. In contrast, the filters corresponding to the sequence r_2 do not exhibit this characteristic (the (b) parts of the figures).
3. The region of the low power unvoiced sound /s/ exhibits little real change in conformity with the sound being produced from a noise-

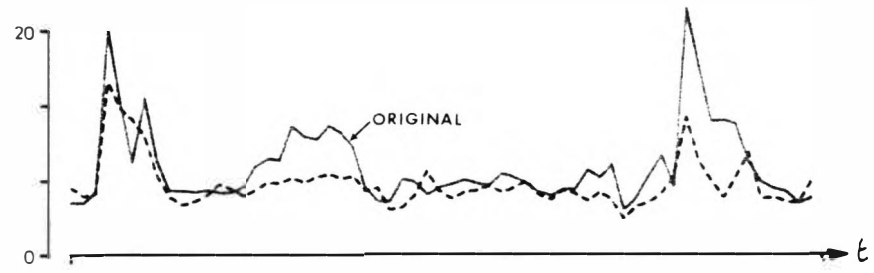


(d) net change in peak factor with time - the difference between the two traces of (c).

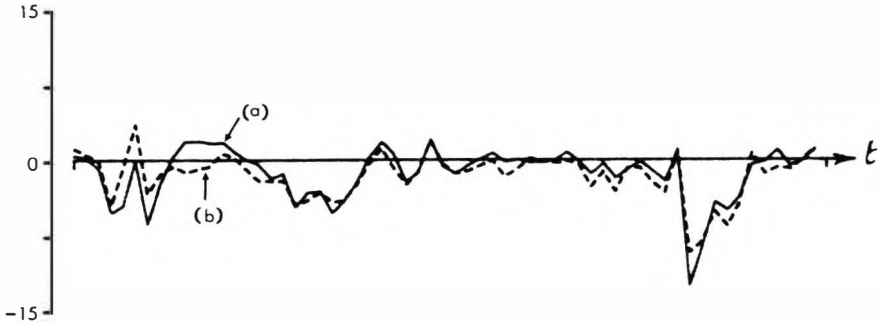
Fig. 5.9: Application of David's method to the word "fast" spoken my male speaker M1: change in signal waveform and segmental peak factor.



(a) original vs FILT1 (sequence r_1)

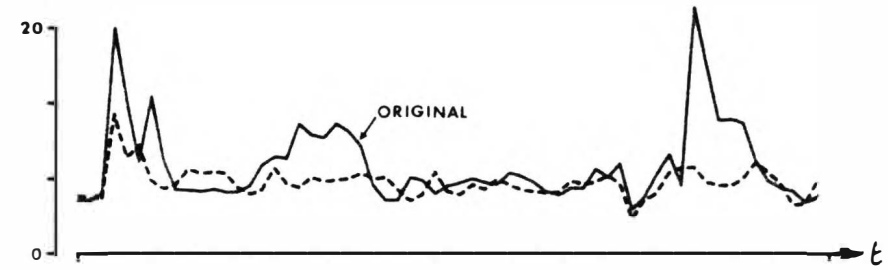


(b) original vs FILT2 (sequence r_2)

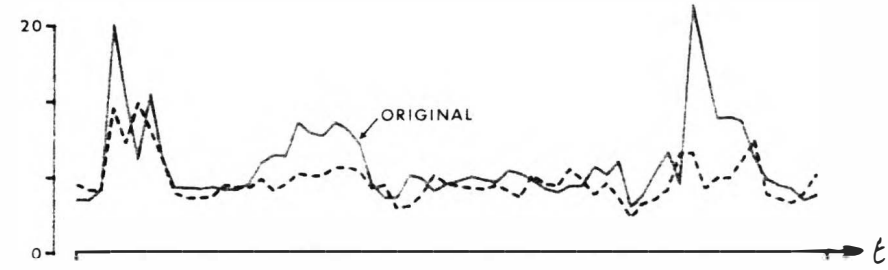


(c) comparison of the net change in PF_{seg} between (a) and (b). Each line represents the difference between the original and result.

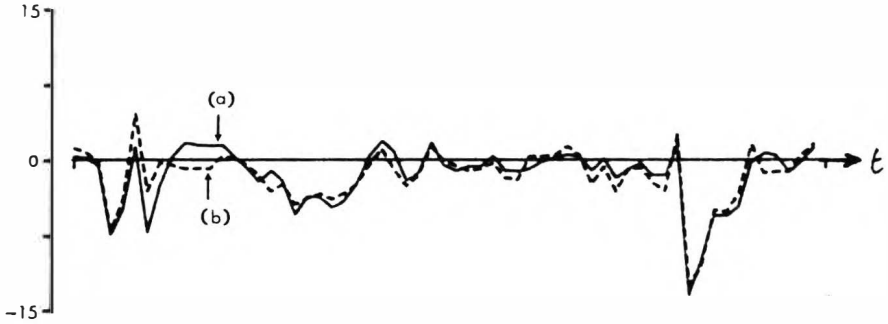
Fig. 5.10: Change in segmental peak factor PF_{seg} versus the defining equation used to generate the filter non-linear phase response. Phase bandwidth= 0-1000 Hz.



(a) original vs FILT3 (sequence r_1)

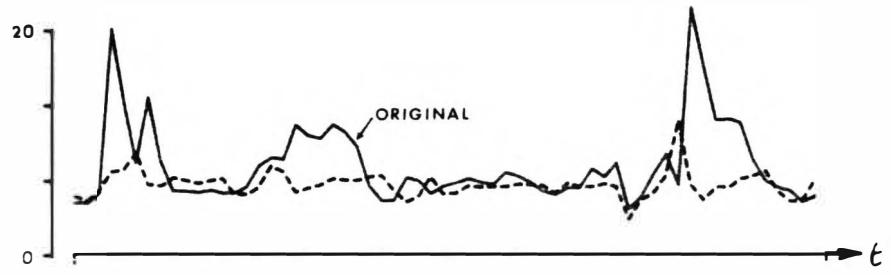


(b) original vs FILT4 (sequence r_2)

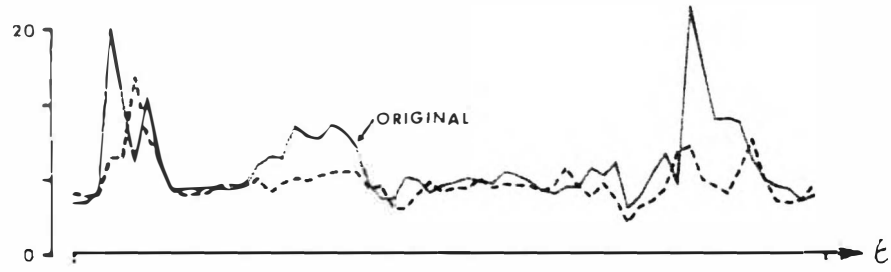


(c) comparison of net change in PF_{seg} between (a) and (b). Each line represents the difference between the original and the result.

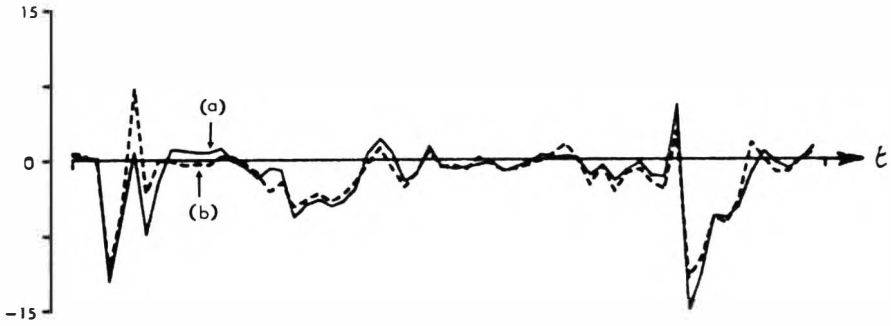
Fig. 5.11: Change in segmental peak factor PF_{seg} versus the defining equation used to generate the filter non-linear phase response. Phase bandwidth= 0-2000 Hz.



(a) original vs FILT5 (sequence r_1)



(b) original vs FILT6 (sequence r_2)



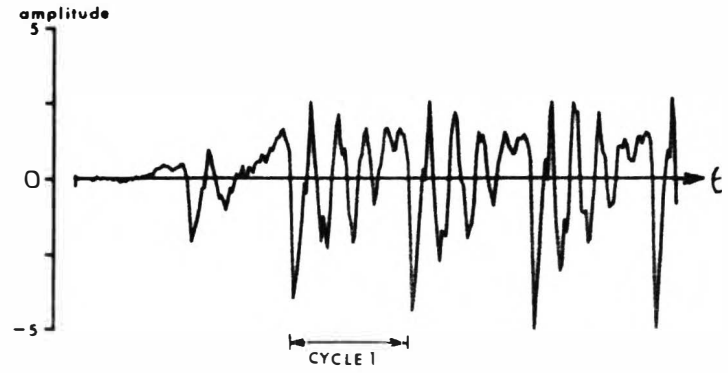
(c) comparison of net change in PF_{seg} between (a) and (b). Each line represents the difference between the original and result.

Fig. 5.12: Change in segemental peak factor PF_{seg} versus the defining equation used to generate the filter non-linear phase response. Phase bandwidth= 0-3000 Hz.

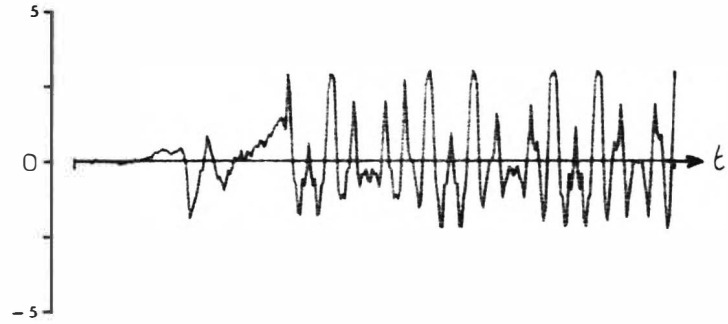
like source with no periodic structure.

3. All the filters used give peak factor reduction in the region of the segment /t/. This phenomenon is peculiar. /t/ is usually considered as being formed from a plosive unvoiced sound and thus should not contain any recognisable periodic structure. The conclusion therefore, is that some elements of voiced structure must exist in this sound.
4. None of the filters used give the same consistent peak factor reduction as David's method over the region of the voiced segment /a/.
5. Part (c) of each of the figures shows more clearly that the filters that use sequence r_2 are more consistent in giving a peak factor reduction, but that the maximum reduction is achieved with the filters formed using sequence r_1.
6. Overlaying Fig. 5.11 (0-2000 Hz) with Fig. 5.12 (0-3000 Hz) confirms that there is no essential difference in the results. Thus, there is no additional gain increasing the phase bandwidth to 3000 Hz.
7. Overlaying Fig. 5.10(c) (0-1000 Hz) with Fig. 5.11(c) (0-2000 Hz) shows that for this example expanding the phase bandwidth out to 2000 Hz causes a decrease in the amount of peak factor reduction obtained - especially in the voiced region /a/.
8. The results of 6. and 7. above show that for this combination of word and phase response (ie original random number sequence) at least, the assumption in section 5.5 recommending a phase response only over the first formant region was justified and that in this case increasing this phase response to 2000 Hz incurs a penalty of reduced performance.
9. Comparing Fig. 5.9(d) with Fig. 5.10(c), David's method is marginally better than the result for FILT2. However, since this filter gives peak factor reduction in other segments of the word, David's method is in general inferior to the filter method.

Fig. 5.13 shows an expansion of the beginning of the vowel segment for the original signal, the result from David's method and the result from FILT2. In both cases the clearly visible pitch period of the original has been destroyed. Fig. 5.14 shows the corresponding analytic envelopes of the signals for the segment marked "cycle 3" in Fig. 5.13(a). Fig. 5.15 shows the amplitude spectrum corresponding to this single cycle for the original and the result from FILT2. The two



(a) original



(b) David's method

(c) using FILT2

Fig. 5.13: The change in behaviour of the signal waveform using David's method and a non-linear phase filter. The beginning of the vowel /a/ is shown.

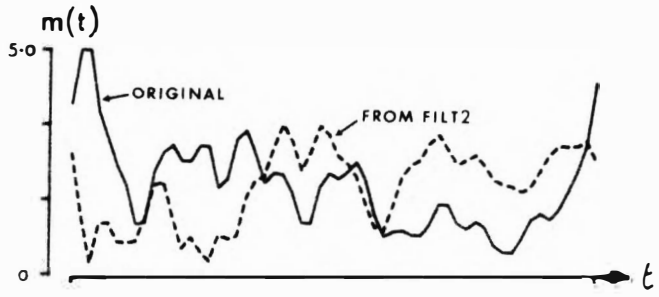
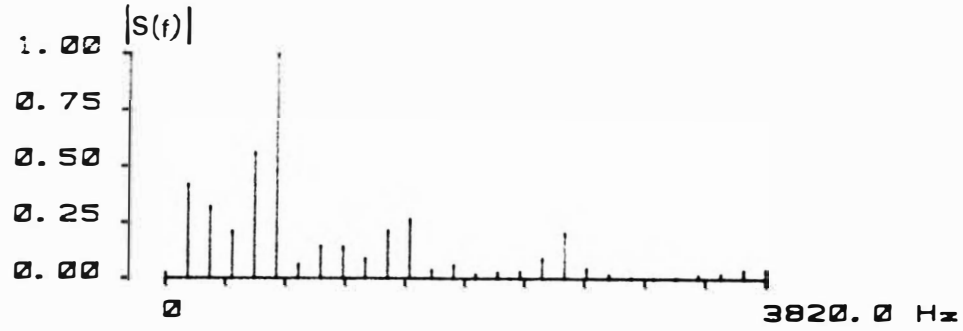
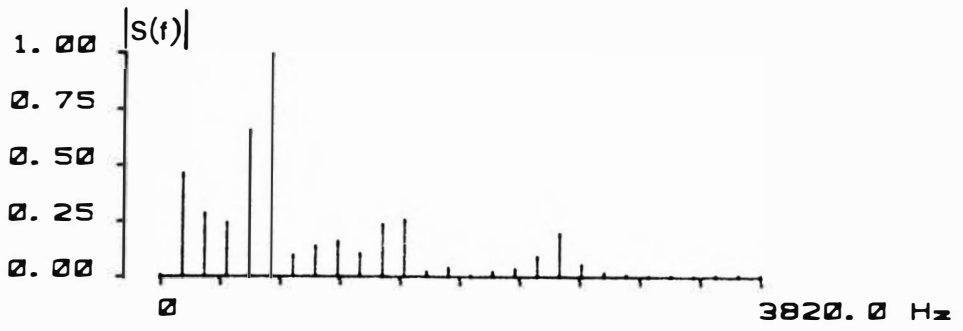


Fig. 5.14: Envelopes of the expansion of "cycle 1" of the same signals in Fig. 5.13.



(a) original



(b) through FILT2

Fig. 5.15: Amplitude spectrum of "cycle 1" of Fig. 5.13 for the original and from FILT2.

spectra are essentially identical confirming that no amplitude distortion has been introduced. The minor differences in some of the harmonic levels is likely to be due to the ripple in the passband of the filter. Table 5.2 gives additional information related to the simulations.

TABLE 5.2 SIGNAL EXTREMA AND AVERAGE PEAK FACTOR FOR THE DIFFERENT METHODS

$\Delta\omega_\theta$ = 100 Hz, ω_o = 100 Hz, N = 299

Δt = 15ms (M=121), P=60

METHOD	s_{max} (V)	s_{min} (V)	dynamic range $s_{max}-s_{min}$	PF_{av}
original	2.603	-4.690	7.293	7.88
David's	3.215	-2.706	5.921	7.16
FILT1	4.170	-5.084	9.254	6.63
FILT2	2.848	-4.055	6.903	6.35
FILT3	3.908	-4.845	8.753	6.39
FILT4	2.666	-3.923	6.589	6.34
FILT5	3.787	-4.254	8.041	6.22
FILT6	3.050	-4.059	7.109	6.31

(Fig. 5.9(a))

The data in Table 5.2 is apparently conflicting when considered in the light of the results presented in Figs. 5.10 - 5.14. The entries for PF_{av} show consistent peak factor reduction over a "long-term" interval which is clearly supported by parts (a) and (b) of Figs. 5.10 - 5.13. However, at the same time, all the filters using sequence r_1 (FILT1, FILT3, FILT5) show consistent increase in signal dynamic range, while those using sequence r_2 show consistent decrease in dynamic range.

Note that David's method gives considerable reduction in the dynamic range of the signal, but the average peak factor reduction over the whole word is poorer than all of the filter approaches. This is simply because David's method was used only over the voiced segment, while the filter methods gave useful reduction in other segments of the word and thus contribute to an overall lower average.

5.6.1 Complex to Real Zero Conversion

The consequences of the result of Appendix 5.0 on the type of zeros required of the signal occurred to the writer at a late stage of the work and time did not permit a more thorough investigation. For this reason, the discussion below deliberately adopts a discussion pose to avoid making unsubstantiated assertions.

For the 3-sinusoid example, a significant result of Appendix 5.0 is that the signal zeros $\{x_k\}$ be entirely real for the analytic zeros $\{z_k\}$ to be evenly spaced in time. For a bandlimited signal, the number of zeros $\{x_k\}$, be they real zero crossings or complex, are, by definition, set by the bandwidth of the signal (equation 4.12). The phase manipulation procedure does not alter the bandwidth of the signal, so the number of zeros is unaltered. The main question is, therefore, can this result be extended to a real signal: ie can the equivalent high order polynomial be forced to possess only real zeros, and what are the consequences for a signal with such a property?

Assuming, for the sake of argument, that such an extension to a real signal is valid, this implies that all the complex zeros of the signal are converted to real zeros. Such a signal would exhibit constant peak amplitude because local minima or saddle points in the signal waveform can only be present if complex zeros exist. Hence, the information contained in the signal will be given solely by the positions (in time) of the real zeros.

Accepting that the phase randomisation procedure is not optimum, then the conversion of complex to real zeros will be incomplete, but such conversion, if it occurs, will be directly observable from an examination of the signal waveform. Fig. 5.16 is an expansion of "cycle 1" of the waveforms of Fig. 5.13. Careful examination of the sample data corresponding to this time period for all the data of Table 5.2 gives the real zero count shown in Table 5.3. In all cases, every phase modified signal shows an increase in the number of real zeros for the period examined. However, different pitch periods gave different increases in the number of real zero crossings.

Note also that the filters with wider phase bandwidth do not give any extra real zeros over the filters with phase bandwidth 0-1000 Hz (FILT1 and FILT2). This supports the original observation above that the important zeros that require manipulation lie in the first formant.

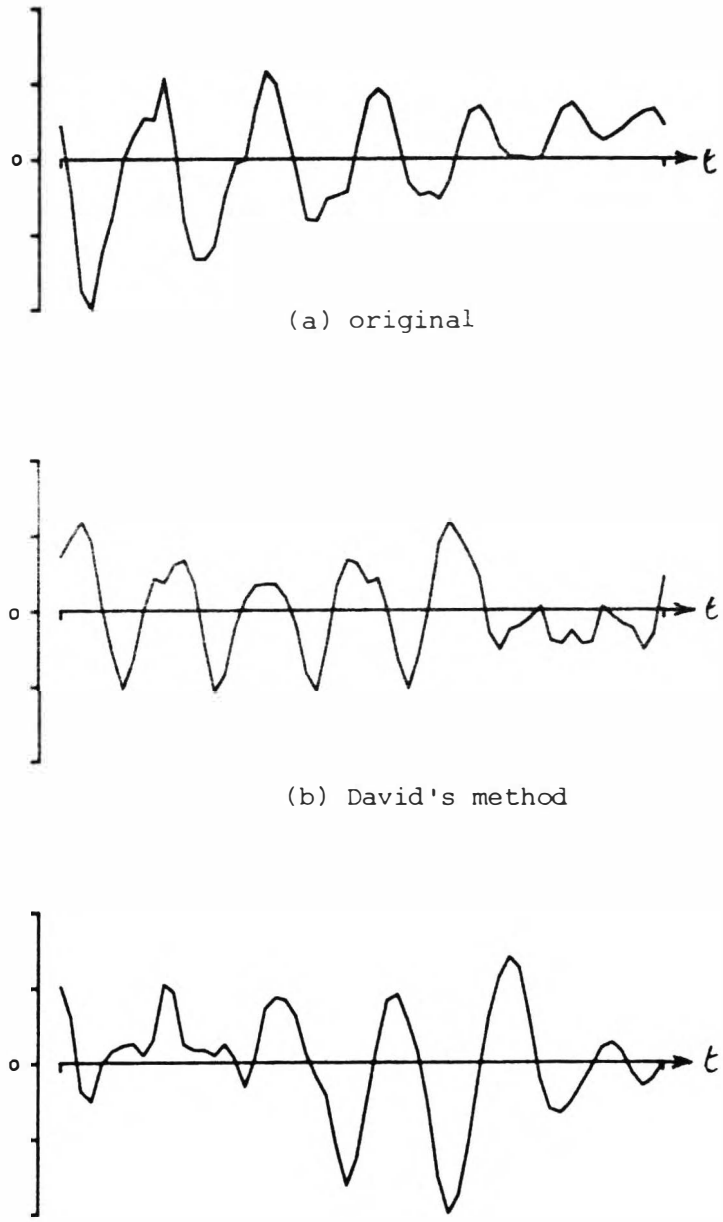


Fig. 5.16: Number of real zero crossings: expansion of "cycle 1" of the signal waveforms of Fig. 5.13. (male speaker).

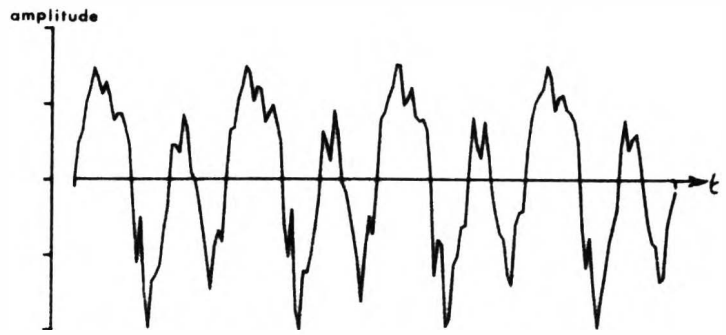
TABLE 5.3: NUMBER OF REAL ZEROS IN A SINGLE PITCH PERIOD FROM A REAL VOICED SEGMENT (MALE M1).
parameters as for Table 5.2

METHOD	no of real zeros
original	10
DAVID's	14
FILT1	13
FILT2	12
FILT3	11
FILT4	15
FILT5	12
FILT6	14

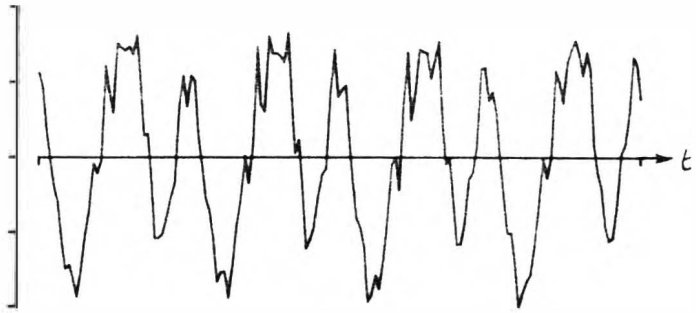
However, for female voices, this phenomenon of complex to real zero conversion is not nearly so obvious. Anticipating the randomised speech generated in section 5.7.1, Fig 5.17 shows four pitch periods of a voiced segment of the female speech F1 and the corresponding segment that results from the application of FILT2 of section 5.6. In this period, the original signal has 16 zero crossings while the randomised version has 20. For the randomised signal produced by FILT1 only 17 zero crossings eventuate for the same segment, but, as it will be shown, the peak factor reduction for FILT2 is better than that achieved for FILT1. That an increase in the number of zero crossings is linked to the peak factor reduction is consistent with a phase manipulation that gives a more even distribution of the analyitc zeros in time.

In general, however, it appears that phase randomisation gives a smaller increase in real zero crossings per interval than achieved for male speech. For now, further discussion of this aspect is held until after all the experiments are described because when considered as a whole, they provide an explanation of this apparent difference in behaviour.

The consequences of retaining the original assumption above that all the zeros of the signal be real are important, but not fully understood. Would such a signal exhibit the lowest peak factor possible, and how close would this value be to the limiting value of π established in section 5.4.2? More importantly, there are known and established techniques for converting complex zeros to real zeros, so



(a) original



(b) result through FILT2

Fig. 5.17: Number of real zero crossings: 4 pitch periods from female speaker F1.

how does this consequence of phase manipulation relate to them?

It is known that if a bandlimited signal is differentiated a finite number of times, all the complex zeros will be converted to real zeros. Differentiation however, involves significant amplitude distortion, but does not explicitly increase the signal bandwidth. However, speech intelligibility and quality deteriorate rapidly after 2 successive differentiations. Indeed, the use of a single stage of differentiation is sometimes used as a crude method of enhancing the intelligibility of speech because this process lifts the mean power of the higher frequencies making the speech sound "brighter".

The process of infinite clipping reduces a signal to a constant amplitude signal wherein only the real zero crossings remain. It is

known that a clipped signal remains highly intelligible [Licklider et al, 1948], but the quality of the signal deteriorates rapidly as the amount of clipping introduced is increased. Furthermore, the process of clipping introduces spectral distortion and increases the bandwidth of the signal.

Hamilton [1985] reconstructed constant amplitude (ie possessing only real zero crossings) voice by forcing the envelope function $m(t)=1$ and using only the instantaneous frequency (equation 4.36). He found that the penalty incurred was an increase in signal bandwidth because the relationship between the envelope $m(t)$ and instantaneous frequency $\omega_i(t)$ had been destroyed. He further showed through arguments related to $\omega_i(t)$ and frequency modulation theory, that removing the amplitude information causes what are essentially negative frequency components that fold back into the spectrum of the signal. The conclusion is that such a signal has suffered spectral distortion which will degrade both the intelligibilty and quality of the signal.

A process of adding a strong bias tone, attributed to Haavik by Sekey [1970], converts the original signal into one containing only real zeros that correspond to the sine-wave crossings of the strong bias tone. Subsequently, Bar-David [1974] used this transformation to prove an implicit sampling theorem. Piwnicki [1983] found the spectra of the output of modulation methods related to the bias tone zero crossings were strictly high pass. One term of the relationship describing the output spectra could be isolated and represented an arbitrary form of amplitude modulation: exactly what type depended on the relationship of the bias tone and the signal. However, the basic conclusion is that since the spectra of the output is high pass, there is an increase in signal bandwidth.

In the absence of more detailed investigation and rigorous proof, the discussion above suggests that a signal modified to possess only real zero crossings will have a bandwidth greater than its original. This would indicate that the original assumption requiring complete conversion of complex to real zeros is too broad, because the phase randomisation process has been demonstrated to maintain the amplitude spectrum of the original signal. In addition, it appears that the effective phase bandwidth required of the filters is less than the signal bandwidth and thus only affects a few of the signal zeros. Thus

there appears to be a finite limit on the allowable number of complex zeros that can be converted to real zeros. What this limit may be is not known, but the results presented here clearly demonstrate an increase in the number of real zeros of the signal.

5.7 PHASE RANDOMISATION OF CONVERSATIONAL SPEECH

In this section, non-linear phase filters are applied to the speech data base established in section 5.3. However, for a given voice, there are several filter parameters that may influence the observed results. For an initial defining random sequence $\{r'_n\}$ (equation(3.19)), two phase responses (sequences r_1 and r_2) can be formed for a fixed $\Delta\omega_\theta$, ω_0 and N. Thus the discussion that follows is partitioned according to the parameter(s) of the filters that are altered.

In addition, with the time length of the speech used here, it is considered that a sufficient number of samples exist to examine the effect of phase randomisation on the signal amplitude probability distribution (pdf) in the manner of Davenport [1962]. Discussion of this aspect is deferred to section 5.7.5. Also, part of the exercise here is listen to the phase randomised speech to find out if any difference can be perceived. Discussion of what the speech sounds like will be deferred to section 5.7.4.

In the sections that follow, the emphasis is on presenting the change in behaviour of PF_{av} and the signal dynamic range of speech originating from two different environments. It is well known that the environment of the speaker modifies the acoustic waveform. Thus one set of speech originates from an anechoic chamber which, it is hoped, will give a signal uncoloured by room acoustics. The other set of speech comes from a domestic living room where room acoustics may have modified the acoustic signal as seen by the microphone. Because the data base is small, no attempt will be made to infer trends or typical behaviour.

The practice adopted in section 5.6 of adjusting the value of s_{av} of the phase randomised signals to be the same as the original is retained here as well.

5.7.1 Filter set used in section 5.6

Using this filter set establishes a base and provides a check on the observed behaviour of the word "fast" of the previous section. In the same manner as Table 5.2, Tables 5.4-5.7 present the results for the four sets of conversational speech.

For the male voices, the essential points of Tables 5.4 and 5.5 are:

1. All the filters give a net lower peak PF_{av} consistent with the word "fast".
2. Negligible, if any, further reduction in peak factor is achieved by using phase badwidth in excess of 1000 Hz.
3. Filters using the r_1 sequence give a consistent decrease in signal dynamic range. Those using sequence r_2 give a consistent increase in dynamic range. This behaviour is the reverse of that observed for the word "fast".
4. The reduction in PF_{av} shown is misleading in that it suggests little change has been made. Fig. 5.18 shows the PF_{seg} plots and the difference (change) in peak factor for the middle 2.5 seconds of the male speech M1. Observe that while some very large peak factor segments have been curtailed, there are regions where increase in peak factor has occurred. The net overall average change is thus small.

For the female voices:

5. A consistent reduction in PF_{av}. However, the net reduction is considerably smaller than achieved for male voices.
6. The original unmodified speech shows that female speech has a generally lower peak factor than male speech.
7. There is no clear pattern of change in signal dynamic range as observed for male speech. An increase in dynamic range appears to be as likely as a decrease. This is especially true of the female F2 who was recorded in her own living room (a natural environment) and is thus considered more representative of the likely behaviour of female speech.
8. Fig. 5.19 (plotted to the same scale as Fig. 5.18) shows the PF_{seg} plots for the the female F1. Comparing these with Fig. 5.18 shows clear differences between male and female speech. The occurrences of very large peak facator segments are absent in female speech.

TABLE 5.4 SIGNAL EXTREMA AND AVERAGE PEAK FACTOR
FOR SPEECH OF MALE M1
$\Delta\omega_\theta$ = 100 Hz, ω_o = 100 Hz, N = 299
Δt = 15ms (M=121), P=60
s_{av} = 0.2495 V

METHOD	s_{max} (V)	s_{min} (V)	dynamic range $s_{max}-s_{min}$	PF_{av}	
original	2.195	-4.783	6.978	6.11	(reference)
FILT1	3.467	-2.697	6.164	5.51	
FILT2	4.224	-2.927	7.151	5.44	
FILT3	3.536	-2.842	6.378	5.53	
FILT4	4.211	-2.849	7.060	5.40	
FILT5	3.552	-2.874	6.426	5.45	
FILT6	3.338	-2.982	6.320	5.43	

TABLE 5.5 SIGNAL EXTREMA AND AVERAGE PEAK FACTOR
FOR SPEECH OF MALE M2
$\Delta\omega_\theta$ = 100 Hz, ω_o = 100 Hz, N = 299
Δt = 15ms (M=121), P=60
s_{av} =

METHOD	s_{max} (V)	s_{min} (V)	dynamic range $s_{max}-s_{min}$	PF_{av}	
original	3.196	-4.702	7.898	6.19	(reference)
FILT1	3.366	-3.208	6.574	5.32	
FILT2	3.817	-3.878	7.695	5.36	
FILT3	3.372	-3.490	6.862	5.35	
FILT4	3.759	-3.862	7.621	5.36	
FILT5	3.501	-3.525	7.026	5.32	
FILT6	3.617	-3.888	7.505	5.36	

TABLE 5.6 SIGNAL EXTREMA AND AVERAGE PEAK FACTOR FOR SPEECH OF FEMALE F1

$\Delta\omega_\theta$ = 100 Hz, ω_o = 100 Hz, N = 299

Δt = 15ms (M=121), P=60

s_{av} = 0.6361

METHOD	s_{max} (V)	s_{min} (V)	dynamic range $s_{max}-s_{min}$	PF_{av}	
original	4.280	-4.639	8.919	5.04	(reference)
FILT1	4.139	-4.286	8.425	4.83	
FILT2	4.421	-4.500	8.921	4.77	
FILT3	4.236	-4.360	8.596	4.87	
FILT4	4.422	-4.969	9.391	4.81	
FILT5	4.370	-4.406	8.776	4.89	
FILT6	4.510	-4.626	9.136	4.82	

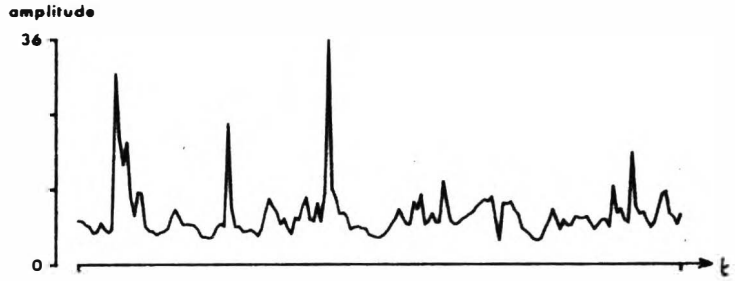
TABLE 5.7 SIGNAL EXTREMA AND AVERAGE PEAK FACTOR FOR SPEECH OF FEMALE F2

$\Delta\omega_\theta$ = 100 Hz, ω_o = 100 Hz, N = 299

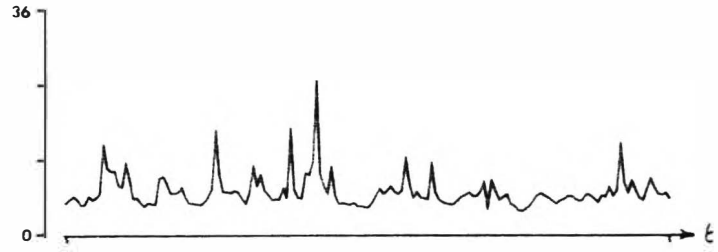
Δt = 15ms (M=121), P=60

s_{av} =

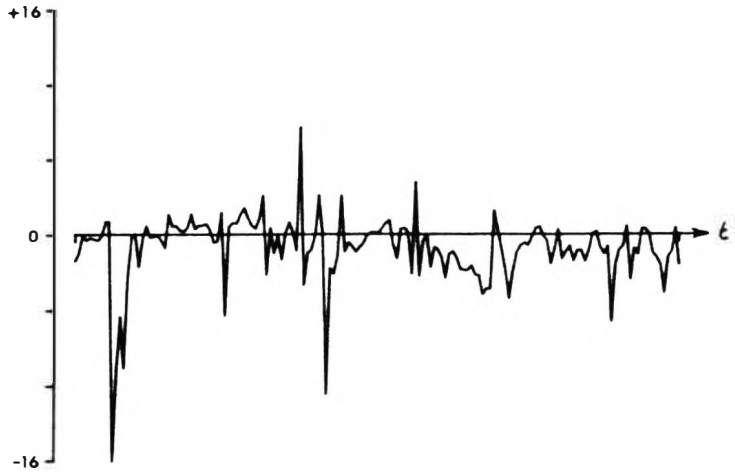
METHOD	s_{max} (V)	s_{min} (V)	dynamic range $s_{max}-s_{min}$	PF_{av}	
original	4.338	-4.419	8.757	5.79	(reference)
FILT1	3.674	-5.510	9.184	5.22	
FILT2	4.196	-4.568	8.764	5.28	
FILT3	3.537	-5.053	8.590	5.22	
FILT4	3.865	-5.558	9.423	5.29	
FILT5	3.693	-5.181	8.874	5.23	
FILT6	4.302	-5.545	9.847	5.34	



(a) original

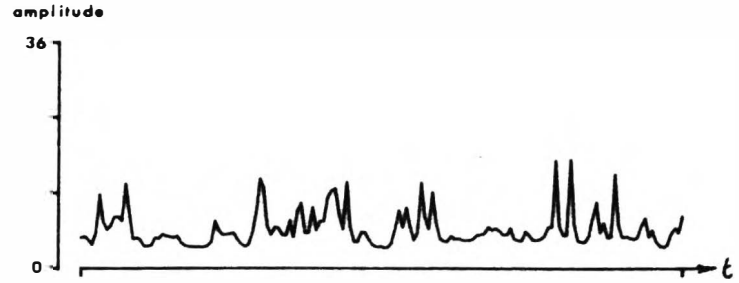


(b) result through FILT2

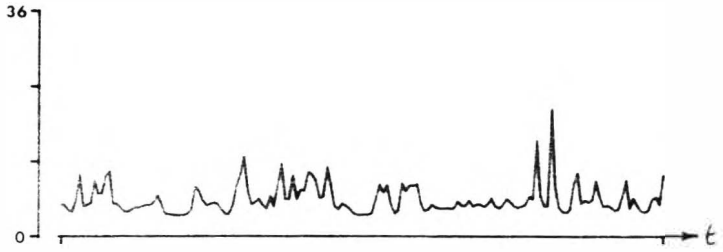


(c) difference (b)-(a)

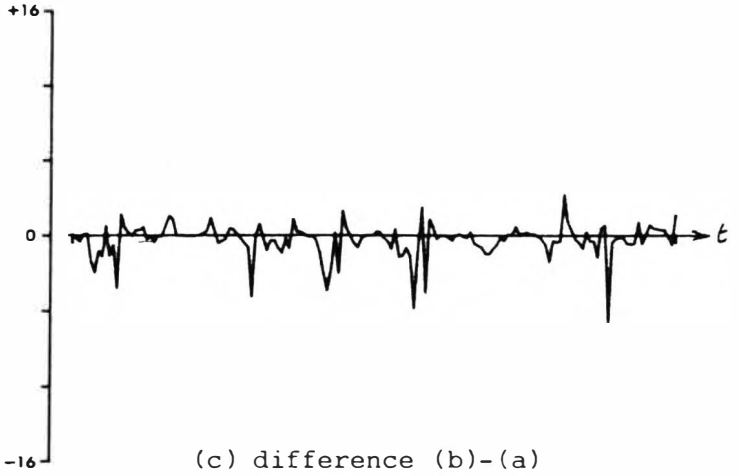
Fig. 5.18: Segmental peak factor of male speaker M1 through FILT2. Middle 2.5 seconds of the utterance shown. t=15 ms; P=60



(a) original



(b) result through FILT2



(c) difference (b)-(a)

Fig. 5.19: Segmental peak factor of female speaker F1 through FILT2. Middle 2.5 seconds of the utterance shown. t=15 ms; P=60. Plotted to the same scale as FIg. 5.18

Further, from Fig. 5.19(c), it is apparent that segments of increase in peak factor occur regularly, while the decreases that are obtained are considerably smaller than those achieved for male speech. The net result is a lower average reduction for females.

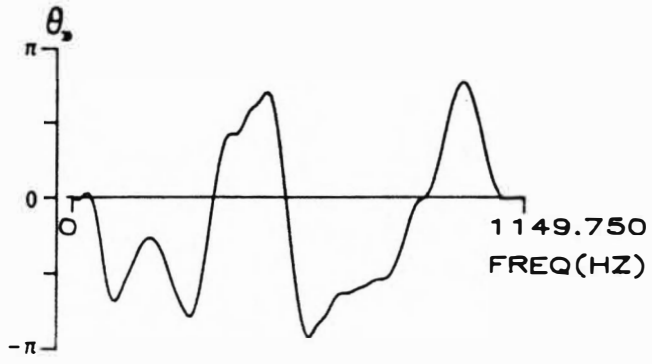
The main conclusions that arise from these results are that female speech appears to be basically "smoother" (less "peaky") than male speech. Peak factor reduction for all speech is consistent, but much less for females. This is proably due to the "smoother" nature of female speech. The phase randomisation process is likely to give an increase in signal dynamic range, especially for female speech. This would suggest that female speech has an inherent larger phase distribution of its harmonics than males, so that manipulation of the phases may in fact cause reduction in the phase distribution leading to an increase in signal amplitude. The physical significance of an increase in dynamic range at the same time as achieving a net peak factor reduction is not clear.

Since neglible additional improvement is achieved for phase bandwidths in excess of 1000 Hz, these extra bandwidths are dropped from any further consideration. Only filters with phase bandwidths of 0-1000 Hz are retained for all future results to follow.

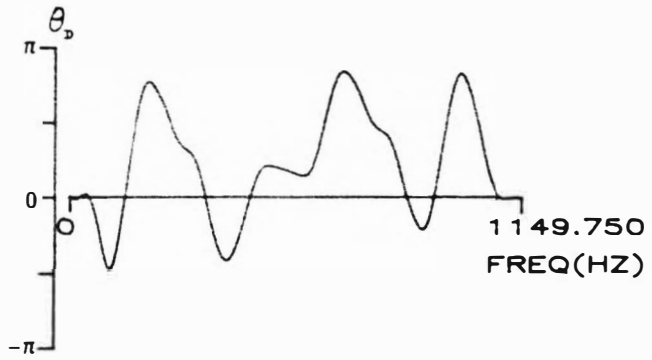
5.7.2 Different Phase Responses

Here, three different filter sets, each with a different initial defining random sequence $\{r'_n\}$ are compared. The phase bandwidth is 0-1000 Hz, and $\Delta\omega_\theta$, ω_o and N are unchanged. Thus the filter sets are {FILT1, FILT1A, FILT1B} corresponding to sequence r_1 and {FILT2, FILT2A, FILT2B} corresponding to sequence r_2. Fig. 5.20 shows the three phase responses of the first filter set.

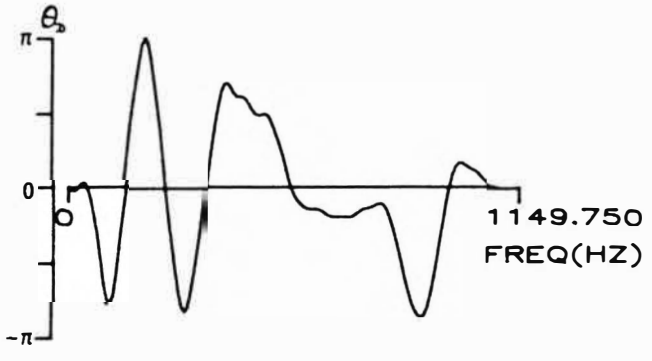
The main issue in this set of results is to examine the sensitivity of different speech to different phase responses. From Tables 5.8-5.11, each phase response gives consistent peak factor reduction for all speakers. There is also noticeable sensitivity to the different phase responses. This is expected since it has already been acknowledged that there are many possible combinations of phases. Some of these phase combinations will give a better peak factor reduction. ALso noticeable is that for the male speakers, the first filter set is consistent in giving a reduction in signal dynamic range while the female voices have the same inconsistent behaviour observed



(a) FILT1



(b) FILT1A



(c) FILT1B

Fig. 5.20: three different phase responses formed from three different defining random sequences (equation 3.19) and using equation 3.20a.

TABLE 5.8 SIGNAL EXTREMA AND AVERAGE PEAK FACTOR FOR SPEECH OF MALE M1: DIFFERENT PHASE RESPONSES COMPARED.

$\Delta\omega_\theta$ = 100 Hz, ω_o = 100 Hz, N = 299

Δt = 15ms (M=121), P=60

METHOD	s_{max} (V)	s_{min} (V)	dynamic range $s_{max}-s_{min}$	PF_{av}	
original	2.195	-4.783	6.978	6.11	(reference)
FILT1	3.467	-2.697	6.164	5.51	
FILT1A	2.860	-3.496	6.356	5.52	
FILT1B	2.714	-3.748	6.462	5.39	
FILT2	4.224	-2.927	7.151	5.44	
FILT2A	3.064	-3.773	6.837	5.33	
FILT2B	3.609	-3.407	7.016	5.42	

TABLE 5.9 SIGNAL EXTREMA AND AVERAGE PEAK FACTOR FOR SPEECH OF MALE M2: DIFFERENT PHASE RESPONSES COMPARED.

$\Delta\omega_\theta$ = 100 Hz, ω_o = 100 Hz, N = 299

Δt = 15ms (M=121), P=60

METHOD	s_{max} (V)	s_{min} (V)	dynamic range $s_{max}-s_{min}$	PF_{av}	
original	3.196	-4.702	7.898	6.19	(reference)
FILT1	3.366	-3.208	6.574	5.32	
FILT1A	3.261	-3.764	7.025	5.34	
FILT1B	3.347	-4.018	7.365	5.29	
FILT2	3.817	-3.878	7.695	5.36	
FILT2A	3.654	-3.127	6.781	5.37	
FILT2B	3.692	-3.260	6.952	5.36	

TABLE 5.10 SIGNAL EXTREMA AND AVERAGE PEAK FACTOR FOR SPEECH OF FEMALE F1: DIFFERENT PHASE RESPONSES COMPARED.

$\Delta\omega_\theta$ = 100 Hz, ω_o = 100 Hz, N = 299

Δt = 15ms (M=121), P=60

METHOD	s_{max} (V)	s_{min} (V)	dynamic range $s_{max}-s_{min}$	PF_{av}	
original	4.280	-4.639	8.919	5.04	(reference)
FILT1	4.139	-4.286	8.425	4.83	
FILT1A	3.919	-5.083	9.002	4.99	
FILT1B	4.235	-4.363	8.598	4.79	
FILT2	4.421	-4.500	8.921	4.77	
FILT2A	4.440	-4.351	8.791	4.90	
FILT2B	3.885	-4.717	8.602	4.85	

TABLE 5.11 SIGNAL EXTREMA AND AVERAGE PEAK FACTOR FOR SPEECH OF FEMALE F2: DIFFERENT PHASE RESPONSES COMPARED.

$\Delta\omega_\theta$ = 100 Hz, ω_o = 100 Hz, N = 299

Δt = 15ms (M=121), P=60

METHOD	s_{max} (V)	s_{min} (V)	dynamic range $s_{max}-s_{min}$	PF_{av}	
original	4.338	-4.419	8.757	5.79	(reference)
FILT1	3.674	-5.510	9.184	5.22	
FILT1A	3.948	-5.877	9.825	5.38	
FILT1B	3.911	-4.720	8.631	5.25	
FILT2	4.196	-4.568	8.764	5.28	
FILT2A	5.213	-4.008	9.221	5.47	
FILT2B	3.897	-4.828	8.725	5.31	

in the previous section.

5.7.3 Different Filter Phase Interval Widths

Here, $\Delta\omega_\theta$ and ω_o (Fig. 5.7) are varied for a constant defining random sequence $\{r'_n\}$ and constant filter length. By virtue of the way in which $\Delta\omega_\theta$ and ω_o are defined in section 5.5, $\Delta\omega_\theta=\omega_o$, so in the results that follow, only $\Delta\omega_\theta$ is indicated. The filter length used is N=2399 because as $\Delta\omega_\theta$ gets very small, it requires an increasing number of points in the filter to keep the passband and stopband ripple at low values. For the the value of N chosen, the passband ripple does not exceed 1dB and the stopband ripple is >65 dB for the filter with the smallest $\Delta\omega_\theta$. In the Tables below, the filter set {FILT1...} uses the sequence r_1 and the filter set {FILT2...} uses the sequence r_2

The motivation for this set of results is based on the development of the required phase response in section 5.5. It was acknowledged that locating the exact harmonic positions is not feasible and further, that these harmonics suffer localised movement (the "teeth" of Fig. 4.10). Thus the ideal in Fig. 5.7(b) that the harmonics fall on the extrema of the phase curve is just an ideal. In practice, the harmonic positions may fall anywhere on the phase curve. Hence, in an attempt to provide more chances of the harmonics falling closer to an extreme point on the phase curve, additional "phase peaks" are added by reducing the width of $\Delta\omega_\theta$. As a consequence, this procedure will alter the effective phase response over the defined phase bandwidth. However, since the defining random sequence is constant, the change in phase response amounts to a steadily increasing compression of the same phase pattern in a given phase bandwidth. For example, Fig 5.8(a) shows the phase response for the original set of filters used over the phase bandwidth 0-3000 Hz. The effect of steadily reducing $\Delta\omega_\theta$ is to compress more of the displayed phase response of Fig. 5.8(a) in the smaller phase bandwidth 0-1000 Hz.

Table 5.12-5.15 indicate the usual parameters. For the male speech M1, with each decrease in $\Delta\omega_\theta$ there is a corresponding decrease in peak factor and then it rises again for the smallest phase interval. This smallest phase interval also corresponds to the biggest reduction in signal dynamic range.

The result for male M2 is rather different: there is effectively no

TABLE 5.12 SIGNAL EXTREMA AND AVERAGE PEAK FACTOR FOR SPEECH OF MALE M1: DIFFERENT $\Delta\omega_\theta$

N = 2399

Δt = 15ms (M=121), P=60

METHOD	$\Delta\omega_\theta$ (Hz)	s_{max} (V)	s_{min} (V)	dynamic range $s_{max}-s_{min}$	PF_{av}	
original		2.195	-4.783	6.978	6.11	(reference)
FILT1	100	3.467	-2.697	6.164	5.51	
FILT1C	50	3.449	-3.108	6.557	5.28	
FILT1D	25	3.446	-4.127	7.573	5.20	
FILT1E	15	2.680	-2.374	5.054	5.27	
FILT2	100	4.224	-2.927	7.151	5.44	
FILT2C	50	3.388	-2.950	6.338	5.41	
FILT2D	25	3.053	-3.775	6.828	5.18	
FILT2E	15	2.550	-3.360	5.910	5.26	

TABLE 5.13 SIGNAL EXTREMA AND AVERAGE PEAK FACTOR FOR SPEECH OF MALE M2: DIFFERENT $\Delta\omega_\theta$

N = 2399

Δt = 15ms (M=121), P=60

METHOD	$\Delta\omega_\theta$ (Hz)	s_{max} (V)	s_{min} (V)	dynamic range $s_{max}-s_{min}$	PF_{av}	
original		3.196	-4.702	7.898	6.19	(reference)
FILT1	100	3.366	-3.208	6.574	5.32	
FILT1C	50	3.401	-3.177	6.578	5.31	
FILT1D	25	3.362	-3.166	6.528	5.30	
FILT1E	15	3.360	-3.106	6.466	5.31	
FILT2	100	3.817	-3.878	7.695	5.36	
FILT2C	50	3.813	-3.856	7.669	5.35	
FILT2D	25	3.851	-3.809	7.660	5.36	
FILT2E	15	3.901	-3.758	7.659	5.35	

TABLE 5.14 SIGNAL EXTREMA AND AVERAGE PEAK FACTOR FOR SPEECH OF FEMALE F1: DIFFERENT $\Delta\omega_\theta$

N = 2399

Δt = 15ms (M=121), P=60

METHOD	$\Delta\omega_\theta$ (Hz)	s_{max} (V)	s_{min} (V)	dynamic range $s_{max}-s_{min}$	PF_{av}	
original		4.280	-4.639	8.919	5.04	(reference)
FILT1	100	4.139	-4.286	8.425	4.83	
FILT1C	50	4.567	-5.125	9.692	4.88	
FILT1D	25	5.126	-5.302	10.428	4.77	
FILT1E	15	5.238	-5.426	10.664	4.75	
FILT2	100	4.421	-4.500	8.921	4.77	
FILT2C	50	4.563	-4.577	9.140	4.89	
FILT2D	25	4.916	-4.637	9.553	4.74	
FILT2E	15	4.792	-5.302	10.094	4.85	

TABLE 5.15 SIGNAL EXTREMA AND AVERAGE PEAK FACTOR FOR SPEECH OF FEMALE F2: DIFFERENT $\Delta\omega_\theta$

N = 2399

Δt = 15ms (M=121), P=60

METHOD	$\Delta\omega_\theta$ (Hz)	s_{max} (V)	s_{min} (V)	dynamic range $s_{max}-s_{min}$	PF_{av}	
original		4.338	-4.419	8.757	5.79	(reference)
FILT1	100	3.674	-5.510	9.184	5.22	
FILT1C	50	3.651	-5.528	9.179	5.22	
FILT1D	25	3.622	-5.469	9.091	5.23	
FILT1E	15	3.657	-5.454	9.111	5.23	
FILT2	100	4.196	-4.568	8.764	5.28	
FILT2C	50	4.213	-4.563	8.776	5.28	
FILT2D	25	4.251	-4.530	8.781	5.27	
FILT2E	15	4.257	-4.536	8.793	5.27	

no improvement in either peak factor reduction or decrease in signal dynamic range as the phase interval is decreased. Whether this behaviour is representative of natural speech (recall male M2 was recorded in a domestic living room) or is an aberration related to the particular speaker is not known because of insufficient data.

The results for the female speech this time are consistent in demonstrating that reducing the phase interval has no effect on the peak factor reduction achieved, but the signal dynamic range show shows a consistent increase.

The main result from the experiments above lie in the noticeable change in the acoustical quality of the phase randomised speech. This is discussed below.

5.7.4 Perceptual Differences in the Phase Randomised Speech

Up to now, the results presented have carefully avoided any reference to the acoustical quality of the phase randomised speech compared to the "original" speech. This is the purpose of this section.

The main result that arises from all the above experiments is that the acoustical quality of the phase randomised speech is directly linked to the phase interval $\Delta\omega_\theta$ used in the filters, and is independent of the filter length N. From the work in Chapter 3, it is known that narrow widths (in frequency - ie the phase widths) require long time lengths to accurately represent the frequency response. Thus, for N constant (but large enough for the narrow phase widths), the resulting behaviour of the speech is linked to $\Delta\omega_\theta$.

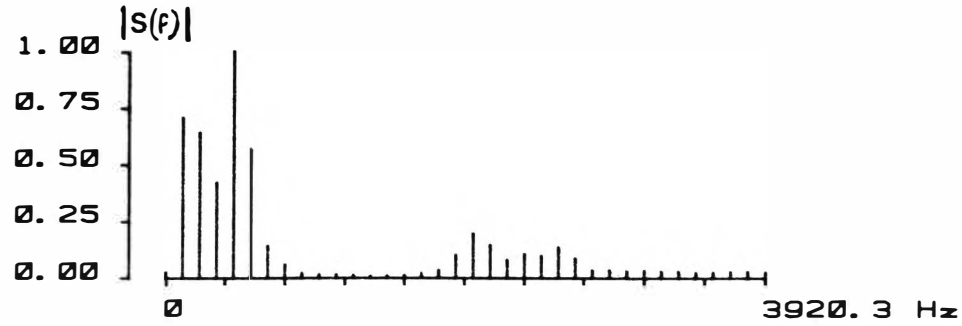
The original filter set used in section 5.7.1 used $\Delta\omega_\theta$=100 Hz. What has not been reported is that filters with $\Delta\omega_\theta$=150 Hz, while giving less peak factor reduction, gave speech that was perceptually indistinguishable from the original, even when listening with headphones. For a phase interval of 100 Hz, it was possible to detect just the slightest change in quality when using headphones: the speech sounded fractionally "brighter". Using speakers (an ordinary listening room environment), no change was apparent. Because the author was familiar with both the phrase and speakers, such an observation may be attributed to familiarity and "knowing what to listen for".

At 50 Hz phase intervals, there was a clearly discernable change

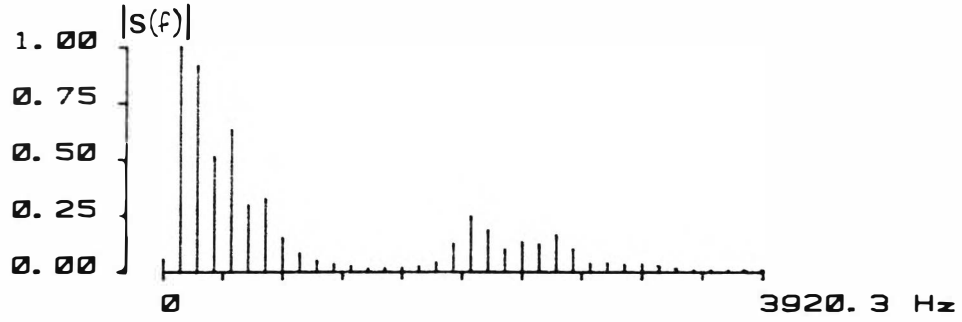
in quality when using headphones: it sounded "brighter". Using speakers, no change was apparent. However, there was no loss in intelligibilty.

At 25 Hz and 15 Hz phase intervals, clear, unambiguous change in speech quality was evident, both with headphones and in an ordinary listening environment. At 25 Hz phase interval, the onset of reverberation was apparent and at 15 Hz phase interval the reverberation had increased to the extent that a secondary "image" signal was clearly distinguishable. Using the speech of male M1 as an example and choosing a single cycle from a voiced portion of the speech at random, Fig. 5.21 shows the amplitude spectra for the same time segment of speech processed by the 25 Hz and 15 Hz phase interval filters compared to the original. This show clearly that the amplitude spectrum has been modified (distorted).

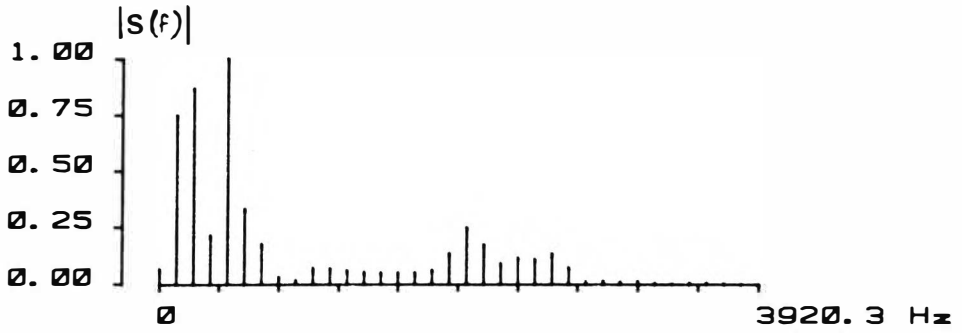
The reason for the 25 Hz and 15 Hz filters causing perceivable amplitude distortion is related to the distribution of energy in the impulse response of the filterand the fact that the signal is speech. In section 3.6 when discussing the design aspects of such random phase filters, it was demonstrated that the single filter could be considered as a composite of many bandpass filters whose bandwidth was the width of the phase interval $\Delta\omega_\theta$. Basic Fourier theory requires that narrow bandwidths require longer time intervals to adequately describe the function. For a filter, this means that significant energy will be distributed over a wider number of points in the impulse response. Fig. 5.22 shows the envelope of a normal linear phase filter and three non-linear phase filters with gradually decreasing phase intervals. Observe the steadily increasing time extent of the energy distribution. For a signal that has constant periodicity (such as several sinusoids), there is no amplitude distortion introduced at all. However, because the speech periodicity is not constant, a long (in time) impulse response will manipulate several pitch periods, all of slighlty different fundamental frequency, at the same time. The result is amplitude distortion. Corroboration of this point is given by Golden [1968] who also pointed out that prolonged impulse response ringing time affected the vocoder performance and introduces reverberation. Thus, if noticeable amplitude distortion is to be avoided, the minimum practical value for phase intervals is in the neighbourhood of 50 Hz.



(a) original

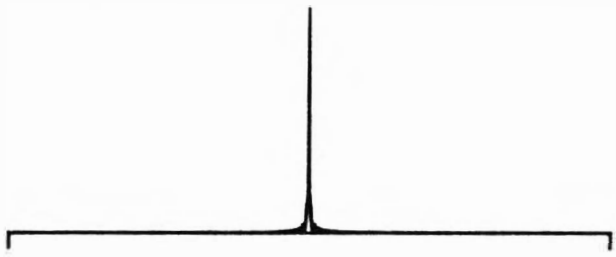


(b) filter phase interval=25 Hz

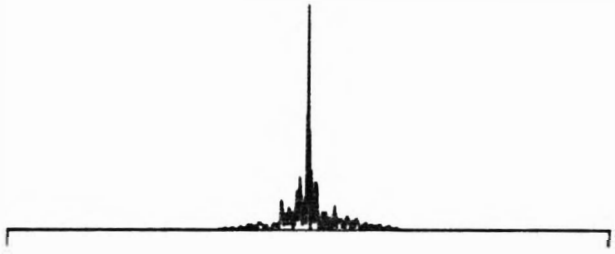


(c) filter phase interval=15 Hz.

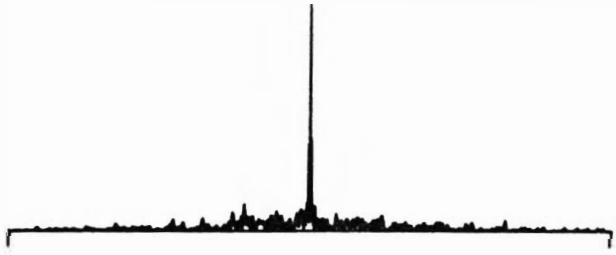
Fig. 5.21: Change in amplitude spectrum for a particular pitch period of male speaker M1 for filter phase intervals of 25 Hz and 15 Hz.



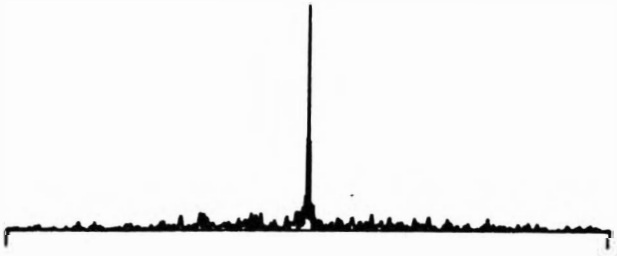
(a) linear phase filter



(b) FILT1 : $\Delta\omega_g$ =100 Hz



(c) FILT1D: $\Delta\omega_g$=25 Hz



(d) FILT1E: $\Delta\omega_g$=15 Hz.

Fig. 5.22: Envelopes of the impulse response h(n) of different non-linear phase filters compared to a linear phase filter. Middle 1000 points of a 2399 point impulse response shown.

5.7.5 The Amplitude Probability Density Function

The one remaining aspect not considered so far is the effect, if any on the amplitude probability density function of speech. Examination of the signal extrema in the tables of data above shows evidence of considerable change, but what of the intermediate values?

Fig. 5.23 shows the pdf plots of the male M1 and female F1 speech for the first filter set data of section 5.7.2 compared to the original.

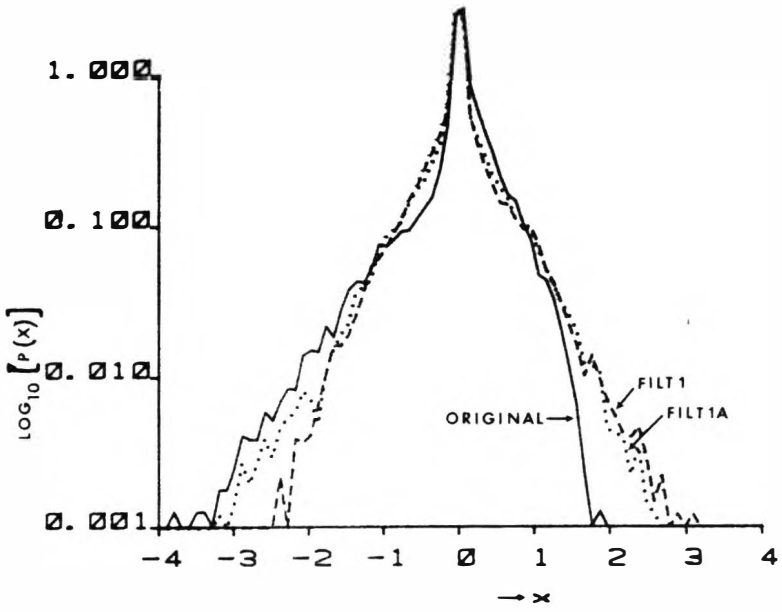
For the male voices, the most obvious feature is that the phase randomisation process tends to shift the pdf symmetrically about the zero point. The region below 0.01 corresponds to the occurrence of high signal amplitudes and it is these samples that are most affected. Above 0.01 there is little real change in the pdf.

In contrast, the pdf's of the unmodified speech are essentially about the zero point. The phase randomisation procedure has almost no effect (female F2) or appears to upset this symmetry and bias it slightly in the negative direction below the probability level 0.01. This is consistent with the observed behaviour of s_{min} in the tables above.

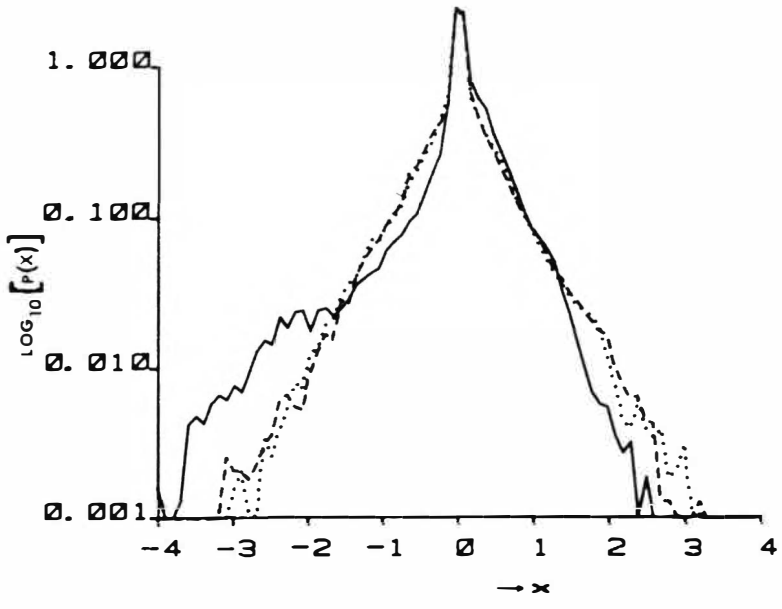
At this point, it is possible to re-assess the apparently conflicting information in the experimental results that show increasing signal extremal values while at the same time gaining a peak factor reduction. The root of the conflict lies in the manner chosen in section 5.3.3 to represent the signal extrema: picking the absolute extremal values gives misleading information. When the pdf of Fig. 5.23(a) was computed, the male speech M1 had 10 samples out of 40,855 with a magnitude greater than 4, but all the phase randomised speech had only 15 samples greater than 3. Similar comments can be made about the female speech. Thus, these few samples represent isolated occurrences (in time) of excessive values that are not representative of the long term signal behaviour. The conclusion is that peak factor reduction does give a decrease in the signal dynamic range, which in the case of female speech is small.

5.8 DISCUSSION AND THE QUESTIONS ANSWERED

Section 5.2.2 proposed five principal questions. Answers to these questions are proposed here, along with concluding comments on the

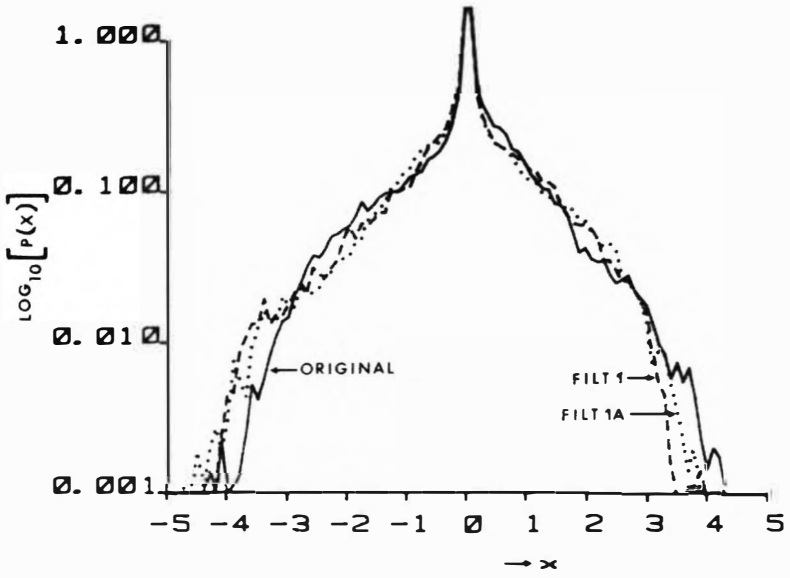


(a) male speaker M1

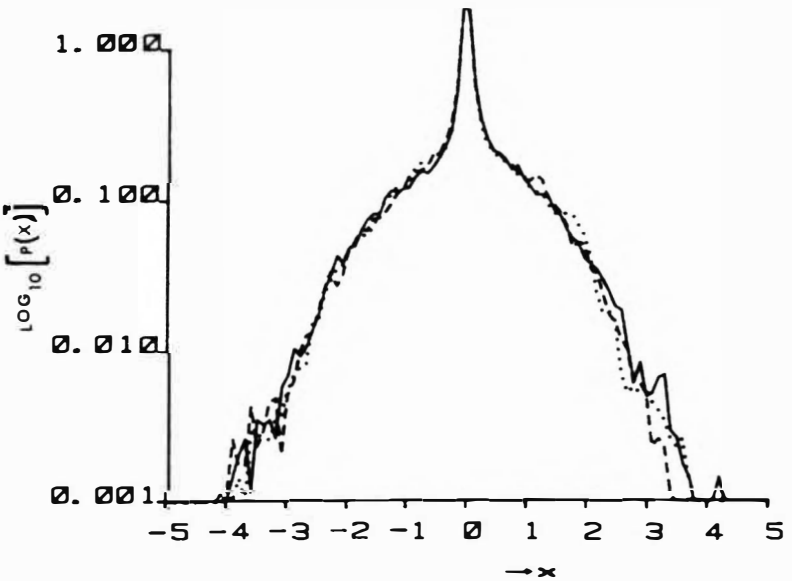


(b) male speaker M2

Fig. 5.23: Amplitude probability density functions for the four speakers from the data used to generate Tables 5.8-5.11.



(c) female speaker F1



(d) female speaker F2

Fig. 5.23 contd: Amplitude probability density functions.

experiments reported above.

The use of FIR non-linear phase filters as a simpler method than the manual Fourier analysis/synthesis has been demonstrated. At the same time, it has been shown that the filter method can produce peak factor reduction in segments of speech that would not be considered using David's method, thereby contributing to a generally better overall reduction. This observation is based on the examination of a single word. The effort required to implement David's procedure for a single word - even with the help of modern graphics terminals - was sufficiently large that it was not considered worthwile attempting to apply it over 5 second speech segments to confirm whether, over a long time interval, the filter method was consistently better than David's method.

Establishing what the actual minimum peak factor may be for a signal in general has not been resolved. A possible irreducible minimum peak factor for any signal of π, has been proposed, based on the observation that this value is the peak factor of a single sinusoid. Since a real signal is composed of many sinusoids, this value of π is suggested as a limiting value that will, in practice, never be achieved.

By considering the analytic zeros of a signal, it has been demonstrated that it is the distribution of these zeros in time that is responsible for the large or small signal amplitudes. Redistributing these analytic zeros exactly evenly in time will lead to the lowest achievable peak factor. However, because the harmonic structure of speech is not constant, even between consecutive pitchs of the signal waveform, and the exact harmonic values cannot be predicted, the filter method is unlikely to achievable the maximum available peak factor reduction. All that can be said is that a particular phase response is likely to give some reduction in peak factor. Exactly what the reduction achieved will be will depend entirely on how well the phase response of the filter matches the harmonic structure of the speaker.

The results of the few experiments reported clearly demonstrate that the nature of male and female speech is completely different and that they are affected in different ways. The plots of pdf for the speech show that male speech speech is influenced to a much greater extent than female speech. Conversely, the harmonic structure of male

speech appears to be far more amenable to phase manipulation than female speech. This is corroborated by the general trend of the results that male speech shows a consistently greater peak factor reduction than female speech.

Recognising that the pitch period of female speech is, in general, twice that of a male, female speech will have far fewer harmonics to represent it. Thus, the order of the polynomial describing it is, in general, half that of male speech. This means female speech has half the number of zeros (signal $\{x_k\}$ zeros or analytic $\{z_k\}$ zeros). Also, natural female speech is identified as sounding "smoother" than a male. Considered from the point of view of its analytic zeros, this implies that they are naturally more evenly distributed in female speech than in male speech. This point is substantiated by the experimental results that showed smaller peak factor reduction for females; the minimal change in the pdf's of female speech and that it appeared to be equally likely that the signal extrema will increase or decrease.

Thus, in section 5.6.1, the issue of complex to real signal conversion for female speech is explained by the fact that there are fewer (important) analytic zeros to manipulate and which are already reasonably evenly distributed in time. This implies that the signal has very nearly the required number of real zeros (whatever that number may be) leaving relatively few complex zeros to be converted to real zeros.

The general conclusion of this re-appraisal of David's early experiment and the consequent characteristcs of speech is that the peak factor of male speech is in general higher than that of female speech and is more readily reduced. The peak factor reductions are relatively modest and thus, as a means of controlling signal dynamic range for use in a communications system (implied in section 5.2.2), it cannot compete with the already well established method of logarithmic companding that is widely used in PCM systems today.

CHAPTER 6

PHASE UNWRAPPING OF SPEECH

6.1 INTRODUCTION

In this final chapter of experimental work, some early and incomplete experiments on using a linear phase FIR filter to achieve bandwidth reduction and subsequent expansion of a signal is described. As a prelude, the subject of speech coding in general, particularly known methods of harmonic compression, are introduced to put the experiments that follow in context.

6.2 SPEECH CODING METHODS

6.2.1 Waveform Coders versus Vocoders

In digital speech coding, there are two fundamental classes of coders: (a) waveform coders and (b) source or analysis/synthesis coders called Vocoders (a contraction of Voice Coders) [Flanagan et al, 1979]. Waveform coders code the signal waveform directly. Vocoders analyse the signal waveform to generate a set of parameters that describe the signal which are then coded and transmitted. For this reason they are also known as source or analysis/synthesis coders.

The principal requirement of the coders is to minimise the bit rate (ie bandwidth) required to transmit the information contained in the signal. Concomitant with this requirement is that the received and decoded speech is perceived as being of similar quality and intelligibility to the original.

Within each class of coders, there are two approaches: time domain and frequency domain. Thus, for waveform coders, the classical time domain approaches are Pulse Code Modulation (PCM) and Delta Modulation (DM) [Steele, 1975], while in the frequency domain, examples are Sub-Band Coding (SBC) [Crochiere et al, 1976; Crochiere, 1977, 1981] and Adaptive Transform Coding (ATC) [Zelinski et al, 1977, 1979]. For vocoders, the classical Channel Vocoder [Dudley, 1939] is a frequency domain technique, while that of Linear Prediction (LPC) of the signal waveform [Atal et al, 1971] is a time domain technique. A more recent approach is Time Encoded Speech (TES) [King et al, 1978; Ching, 1983] which transmits coded shape descriptors for successive extended

segments of the speech waveform.

It is not the purpose here to describe all the myriad approaches to coding speech. The recent monograph of Jayant and Noll [1984] covers in an integrated and detailed manner the concepts and algorithms of waveform coding in general. The text of Markel and Gray [1976] integrates the issues associated with linear prediction of speech. In the context of what follows, there are five principle aspects of speech coding that are important:

(a) the majority of coding methods achieve bit-rate (bandwidth) reduction in the transmitted signal by exploiting redundancies in the original speech signal in different ways, but without explicitly using methods of harmonic compression of the speech signal.

(b) waveform coders in general give better quality decoded speech than vocoders. Vocoders tend to produce speech with a synthetic quality and their performance is often talker dependent [Flanagan et al, 1979].

(c) the type of speech coder (waveform or vocoder), its complexity (hardware) and transmission bit-rate are inter-related. In general, vocoders have a higher complexity than waveform coders - Table 6.1.

(d) waveform coders in general require higher transmission bit-rates than vocoders to achieve a given quality. But, at the same time, higher complexity coders allow the same quality to be achieved at lower bit-rates - Table 6.2.

(e) the corollary to the statement in (d) above is that to maintain a given speech quality at lower bit rates, the complexity of the coder required increases. This suggests that there is a boundary (in bit-rate) where the most suitable speech coder will change. Indeed, with the present knowledge available, the quality of waveform coders diminishes significantly below 16 kbits/s, but vocoders achieve a maximum quality at around 4.8 kbits/s with no worthwhile improvement as the bit-rate is increased [Flanagan et al, 1979]. Fig 6.1 illustrates this aspect.

TABLE 6.1 HARDWARE COMPLEXITY OF DIFFERENT CODING METHODS RELATIVE TO ADAPTIVE DELTA MODULATION. [after Flanagan et al, 1979]

Relative Complexity	Coder	
1	ADM:	adaptive delta modulation
1	ADPCM:	adaptive differential PCM
5	SUB-BAND:	sub-band coder (with CCD filters)
5	P-P ADPCM:	pitch predictive ADPCM
50	APC:	adaptive predictive coder
50	ATC:	adaptive transform coder
50	ΦV:	phase vocoder
50	VEV:	voice-excited vocoder
100	LPC:	linear predictive coefficient (vocoder)
100	CV:	channel vocoder
200	ORTHOG:	LPC vocoder with orthogonalized coefficients
500	FORMANT:	formant vocoder
1000	ARTICULATORY:	vocal-tract synthesiser; synthesis from printed English text.

TABLE 6.2 BIT-RATE REQUIRED BY DIFFERENT CODING SCHEMES TO ACHIEVE TOLL QUALITY TRANSMISSION. [after Flanagan et al, 1979]

Coder	kbits/s
Log PCM	56 **
ADM	40
ADPCM	32
SUB-BAND	24
Pitch Predictive ADPCM	24
APC, ATC, ΦV, VEV	16

** the 56 Kbit/s rate is in the USA only. The CCITT standard is 64 Kbits/s.

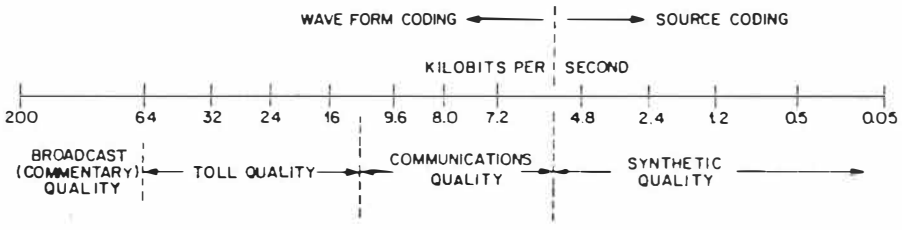


Fig. 6.1: Coder type and quality vs bit-rate. [after Flanagan et al, 1979]

6.2.2 Frequency Division Coders

There exist coding methods - both waveform and vocoder - that achieve bit rate reduction by compressing (dividing) the spectrum of the original speech signal by a factor q>1 (q normally an integer) in such a manner that it only occupies a fraction B/q (B the signal bandwidth) of the original signal bandwidth.

The classical early techniques are those of Marcou [Marcou et al, 1955], the Vobanc [Bogert, 1956], the CODIMEX system [Daguet, 1963] which was a development of the earlier work of Marcou; the Phase Vocoder [Flanagan et al, 1966] and the Analytic Signal Rooter (ASR) [Schroeder et al, 1967] which was an improvement on the earlier Vobanc. The three common factors between all of these methods, are that they are

(a) all in the vocoder class
(b) all frequency domain techniques and
(c) they all extract and manipulate, in one form or another, the analytic or instantaneous envelope m(t), instantaneous phase $\phi(t)$ or instantaneous frequency $\omega_i(t)$:

There are two distinct aspects of the spectrum of speech signal which are important in perception as shown in Fig. 6.2 - the formant structure (the resonances of the vocal tract) and the pitch or fundamental frequency of the voice (the spectral line spacing). The different methods of frequency division modify each of these spectral features in different ways. The Vobanc, CODIMEX and ASR techniques are all wideband techniques and attempt to scale the spectral envelope directly - ie the formant structure. The phase vocoder is a narrow band technique that seeks to scale the pitch harmonics directly. The terms wideband and narrow band are qualitative: the wideband

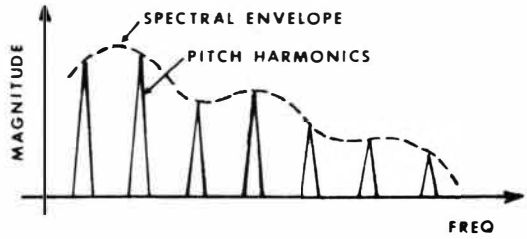


Fig. 6.2: Principal spectral characteristics of speech of concern in frequency division and multiplication.

techniques treat the speech as single contiguous band [Marcou et al, 1955], or as 3 or 4 narrower band signals (Vobanc, CODIMEX, ASR techniques). The phase vocoder separated the speech into 30 bands.

Re-writing, for the sake of clarity, the analytic relationships, the analytic signal $\psi(t)$ is

$$\psi(t) = s(t) + j\hat{s}(t)$$

$$= m(t)e^{j\phi(t)} \qquad (6.1)$$

where: instantaneous amplitude (envelope) is

$$m(t) = \left(s^2(t) + \hat{s}^2(t)\right)^{\frac{1}{2}} \qquad (6.2)$$

instantaneous phase is $\phi(t) = \tan^{-1}\left(\frac{\hat{s}(t)}{s(t)}\right)$ (6.3)

instantaneous frequency is $\omega_i(t) = \frac{d\phi}{dt}$ (6.4)

so that the real signal is given by

$$s(t) = m(t)\cos[\phi(t)] \qquad (6.5)$$

In the absence of modern computers, the early methods of Marcou, CODIMEX and Vobanc used SSB-SC modulation techniques to generate the signal of equation (6.5) - the penalty being that the signal is no longer at baseband and thus requires demodulation at the receiver.

The principle of Marcou's method, shown in Fig. 6.3, generated a constant amplitude compressed signal

$$s_{1/q}(t) = \cos\left[\frac{\phi(t)}{q}\right] \qquad (6.6)$$

and he showed that constant amplitude reconstructed speech was highly intelligible. He further showed that the signal m(t) contained information about the excitation of the original signal. The Vobanc method, shown in Fig. 6.4, used bandpass filters to separate the main formants of speech into three bands about 1000 Hz wide and subsequently generated three separate versions of equation (6.6), but in contrast to Marcou's approach, the Vobanc incorporated the amplitude information m(t) of equation (6.5), but did not scale it. Both systems attempted to halve the signal bandwidth - ie q=2 above. However, when the narrow band filters (shown dotted in Fig. 6.3; the "B" filters of the encoder in Fig. 6.4) were inserted, both systems demonstrated considerable impairment in quality with the introduction

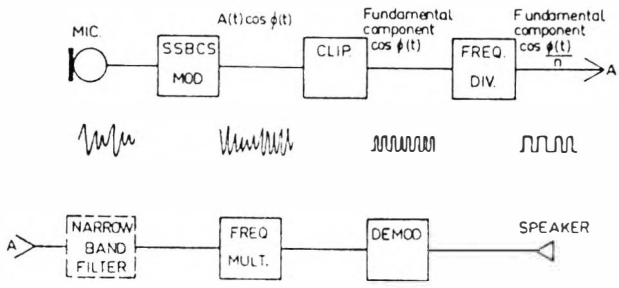
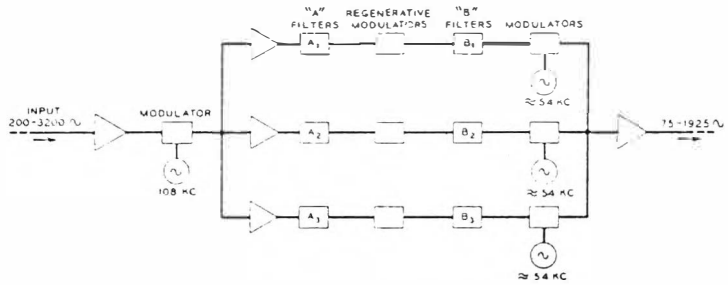
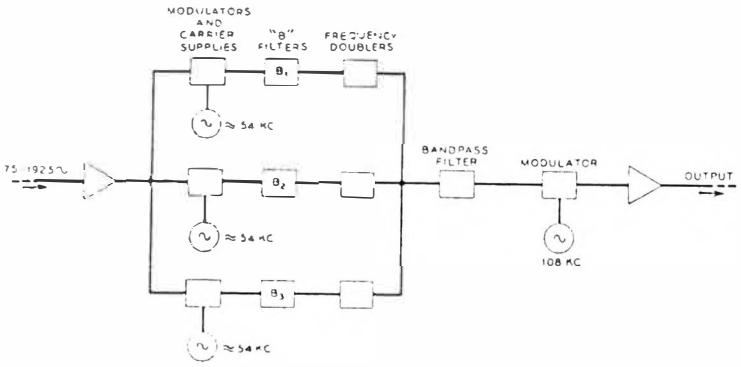


Fig. 6.3 The principle of Marcou and Daguet's system.



(a) encoder (divider)



(b) decoder (multiplier)

Fig. 6.4: Principle of the VOBANC. The filters are bandpass: the "A" filters select the three principle formants. The "B" filters have half the bandwidth of the corresponding "A" filter. [after Bogert, 1956]

of irregularly timed "burbles".

It is now known that both m(t) and $\phi(t)$ (or more usually $\omega_i(t)$ because it is more tractable and easier to compute) are important to the eventual quality of reconstructed speech [Flanagan et al, 1966; Flanagan, 1980a, Flanagan et al, 1980b; Hamilton, 1985]. A careful investigation of the processes of division and multiplication [Bogner 1965; Bogner et al, 1969] for signals that had narrow bandwidth relative to a mean (carrier) frequency, showed that such operations did in fact achieve bandwidths that were essentially 1/q (or q) times the original, but that the process of division or multiplication only accurately scaled the dominant harmonic of the signal band and left the harmonic spacing largely unaffected. ie - the mean instantaneous frequency "captures" the division or multiplication. Thus, in the absence of any amplitude information m(t), any bandlimiting of $\phi(t)$ (or $\omega_i(t)$) introduces phase ambiquities (discontinuous jumps in phase) which cannot be resolved. It is these ambiguities which result in the "burbles".

The CODIMEX and ASR techniques were considerably more successful because they used both the amplitude and phase information. The CODIMEX technique was built using analog hardware involving modulation to generate (and subsequently expand) the compressed signal for each of three bands:

$$s_{1/8}(t) = m^{1/8}(t)\cos\left[\frac{\phi(t)}{8}\right] \tag{6.7}$$

The division by 8 was achieved with three successive applications of division by 2. The ASR technique was a radical departure from the methods so far described. Using division by 2, and separating the speech into four bands, it was shown modulation was not required and that the the half-band signal for each channel is given by

$$s_{1/2}(t) = \left(\frac{1}{2}\right)^{\frac{1}{2}}\left[m(t)+s(t)\right]^{\frac{1}{2}} \tag{6.8}$$

and the expanded signal by

$$s_2(t) = s^2_{1/2}(t) - \hat{s}^2_{1/2}(t) \tag{6.9}$$

The second major difference was the simulation of this system on a computer. Fig. 6.5 illustrates the operations for one channel.

All the systems described so far considered only the spectral

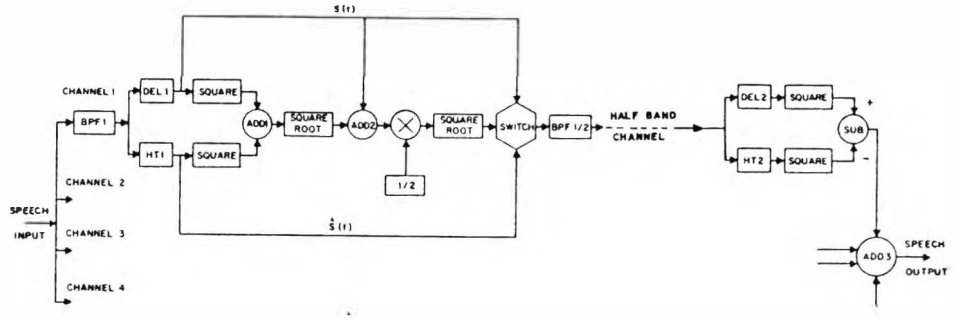
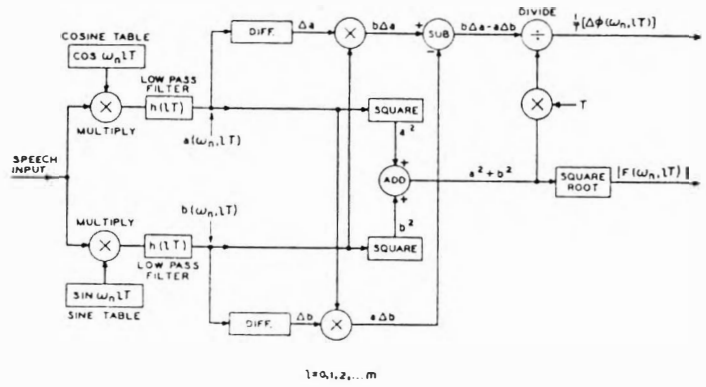


Fig. 6.5: Principle of the Analytic Signal Rooter.
[after Schroeder et al, 1967]

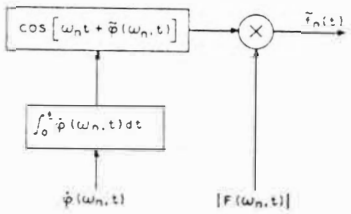
envelope. One method that works on the pitch harmonics is the Phase Vocoder. In describing their invention, Flanagan et al [1966] showed that, in the limit of sufficiently narrow bandpass filtering of the speech that effectively separated the individual pitch harmonics, the short time Fourier magnitude and phase of each narrow band channel output was equivalent to a continuous description of the envelope m(t) and phase $\phi(t)$ of the analytic signal for the particular harmonic. In practice, they used 30 channels. In addition, the system was simulated on a computer, where they used $\omega_i(t)$ instead of computing $\phi(t)$ because it is simpler to generate. Fig. 6.6 shows the operations performed for one channel. Some years later, a digital version of the phase vocoder, using the FFT, was demonstrated [Portnoff, 1976].

More recently, another narrow band frequency domain scaling technique that uses block FFT processing has been described [Flanagan et al, 1980c] with subsequent improvement and efficient implementation [Malah et al, 1981]. Yet another recent frequency domain approach is that of Seneff's [1982], where FFT's again play a central role, but most importantly, the method avoids the need for explicit pitch extraction that is the hallmark of most of the frequency scaling methods.

All the systems described so far are, as already mentioned, classified as vocoder systems. The only known harmonic scaling waveform coder and which is a time domain technique, is the recent Time Domain Harmonic Scaling (TDHS) method, described by Malah [1979] and implemented in real time [Cox et al, 1983]. Good quality reconstructed speech for this method with division by 2 has been



(a) coder



(b) decoder

Fig. 6.6: Principle of the Phase Vocoder - computations required for each of 30 channels. [after Flanagan, 1966]

reported, with some degradation evident with division by 3. The principle of the technique, illustrated in Fig. 6.7, relies on the introduction of pitch information into the frequency scaling operations which are based on a better understanding of the ramifications of the short-time Fourier transform.

6.3 PHASE UNWRAPPING

The method of harmonic compression to be illustrated here uses the coefficients of a linear phase FIR filter. Based on the classification of coder types above, this method can be described as a waveform coder, operating in the time domain. The VAX FORTRAN subroutines that implement the procedures to be illustrated are listed in Appendix 7.

6.3.1 Introduction

The motivation for the experiments stems from a re-interpretation of the implicit linear phase term exp[-jωT(N-1)/2] present in the

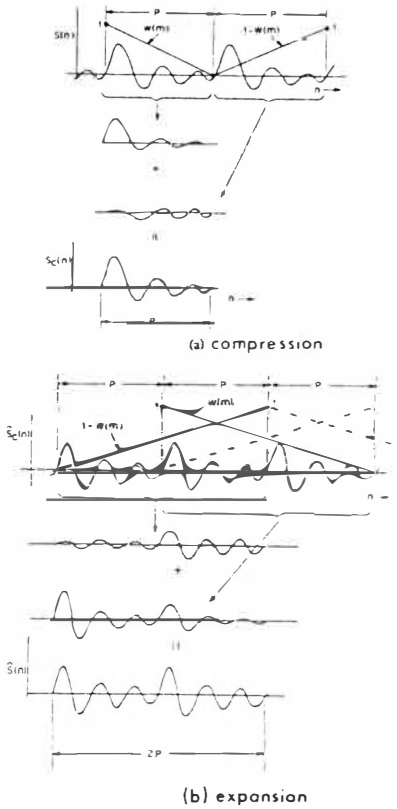


Fig. 6.7: Principle of the Time Domain Harmonic Scaling procedure. [after Cox et al, 1983]

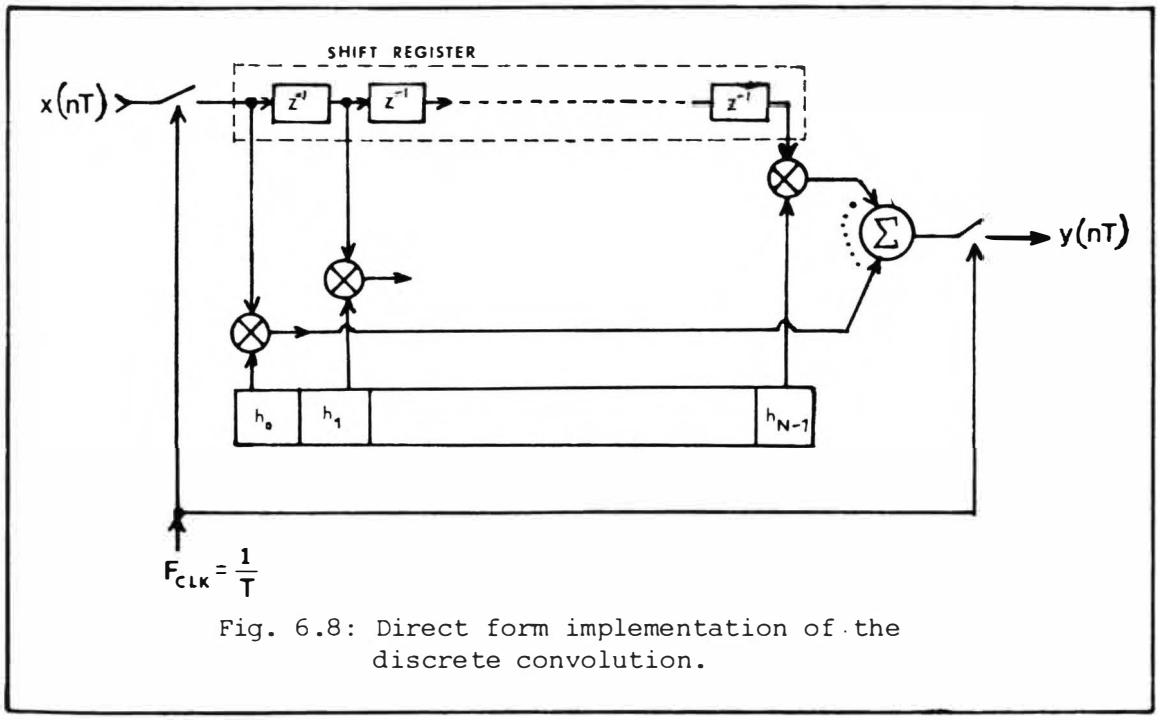


Fig. 6.8: Direct form implementation of the discrete convolution.

output of a convolution using a causal FIR filter. Writing the convolution process as

$$y(nT) = \sum_{k=0}^{N-1} h(kT)x(nT-kT) \qquad (6.10)$$

where h(kT) are the coefficients of an N-point linear phase FIR filter, x(nT) is the input signal and T the sampling interval, Fig 6.8 depicts the usual operation of equation (6.10). From the work in Chapter 2, the coefficients of such a filter are purely real and even symmetric: $h(k) = h(k-\ell)$, $\ell=0,\ldots,(N-1)/2$ and the filter therefore has zero phase deviation (equation (2.8)) relative to the linear phase base (ie the centre of symmetry of the coefficients). Assuming the magnitude of the filter frequency response is unity in the passband, then the Fourier transform of equation (6.10) is

$$|Y(\omega|e^{j\Phi(\omega)} = e^{-j\omega T(N-1)/2} \; |X(\omega)|e^{j\phi(\omega)} \qquad (6.11)$$

so that the output signal phase is

$$\Phi(\omega) = \phi(\omega) - \omega T(N-1)/2 \qquad (6.12)$$

which is simply another way of expressing the time shifting property of the Fourier transform

$$y(nT) = x(nT-n_0T), \quad n_0 = (N-1)/2 \qquad (6.13)$$

Now, considering a single frequency ω_1 and for simplicity, harmonic division by 2, if the output signal frequency is to be $\omega_1/2$, then the rate of change of phase in the output is half that of the input. This requires in equation (6.12) that the phase $\Phi(\omega)$ is half the input. However, without resorting to frequency domain techniques, only the convolution operation of equation (6.10) is available. This leaves the term $\omega T(N-1)/2$ as the only one available for manipulation. Variation of this term implies a shift in the position of the impulse response coefficients of the filter so that the peak of the main lobe no longer stays in the centre (at (N-1)/2) of the sequence. Since h(kT) is periodic in N, such a shift is actually a rotation of the N filter coefficients. Fig. 6.9 depicts the concept.

This variable delay can be represented by defining

$$\zeta(r) = -\omega Tr \quad , \quad r=0,\ldots,N-1 \qquad (6.14)$$

and $\zeta(r)$ has the form shown in Fig. 6.10, where, for T constant, the slope obviously depends on ω. The implication of equation (6.14) is

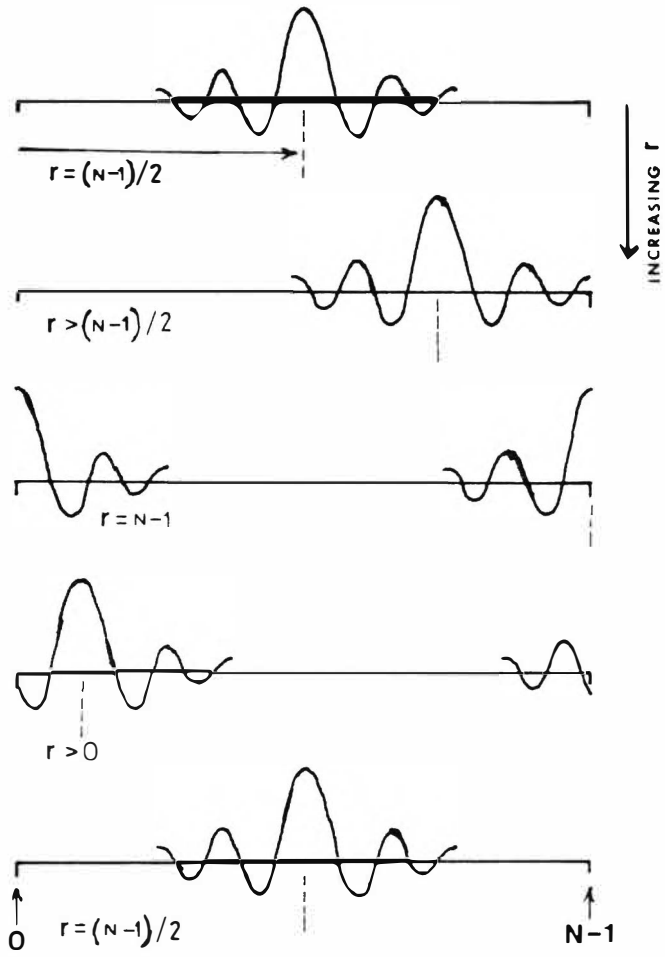


Fig. 6.9: Change in inherent filter delay by shifting the peak of the impulse response. r is defined in equation (6.14).

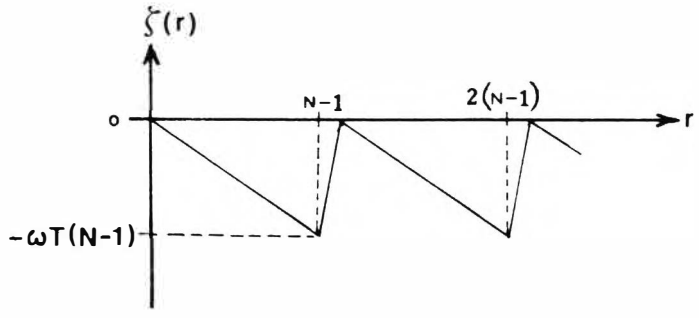


Fig. 6.10: Pattern in change in delay of filter as a function of the offset r of the peak of the impulse response.

that a time-varying filter is being applied to the input signal x(nT) so that the output phase in equation (6.12) will now alter with time (index r).

The principal question is then: what are the ramifications of a periodic variation (rotation) of the filter coefficients?

6.3.2 The idea applied

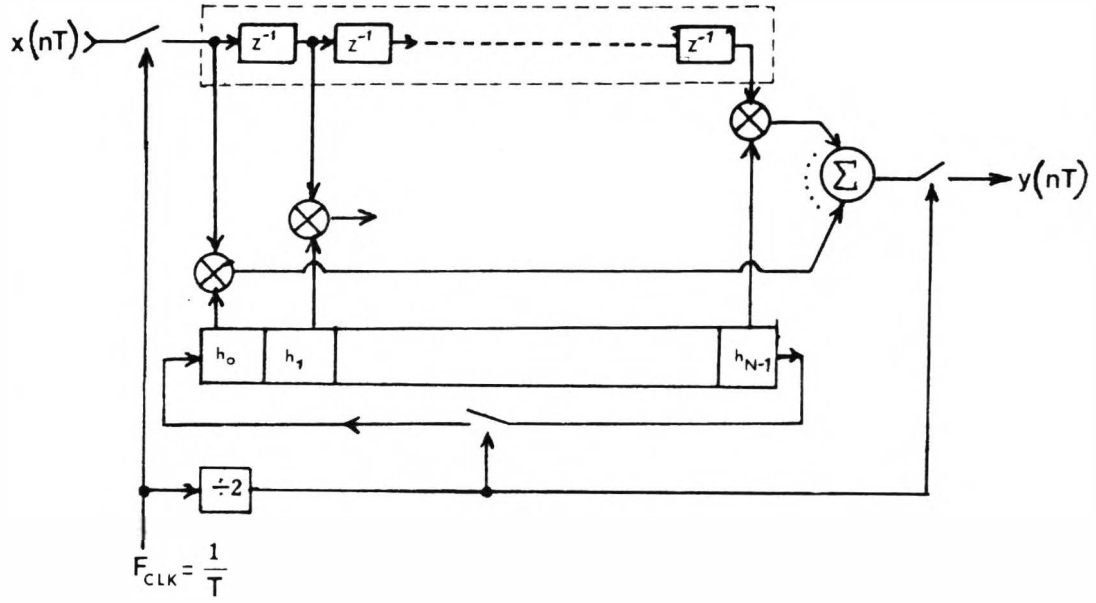
Fig. 6.11 depicts the adaptation of the basic convolution process of equation (6.10) that gives a division by 2. The filter coefficients h(k) and its required frequency response are computed at the input signal sampling rate. At the start, the filter coefficients are symmetrical in the shift register. Subsequently, for every two input samples shifted in, the filter coefficients are rotated one sample to the right and the output computed.

This procedure described above divides the input signal frequency by 2 and at the same time reduces the sampling rate of the output data sequence by 2 - ie the output frequency is $\omega/2$ whose sampling rate is $F_s/2$, where F_s is the original, input data sampling rate. Thus, Fig. 6.12 that shows the signal waveforms and spectra for a single sinusoid has plotted identical time lengths of both the input and output waveforms. The frequency used in Fig. 6.12 is an exact multiple of the filter length - ie

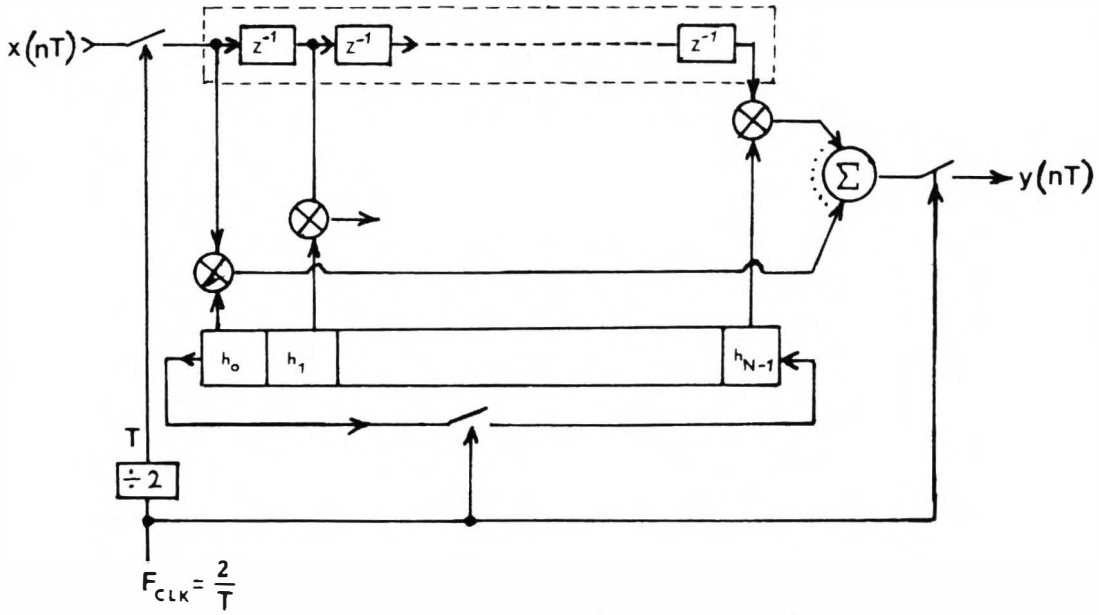
$$\omega = n/NT \ , \quad n=1,.... \tag{6.15}$$

Signals composed of several sinusoids, all exactly harmonicly related to the filter period undergo exact frequency scaling.

However, when the signal frequency is not an exact multiple of the filter length, the output displays regular discontinuities as shown in Fig. 6.13 which coincide with the point at which the peak of the impulse response wraps from the end of the register back to the beginning each time. In addition, an expansion of the region of the discontinuity in Fig. 6.14 shows that the signal develops "kinks" in the few samples before and after the main lobe of the filter wraps around. This phenomenum is related to filter bandwidth. The first filter used to generate the trace (a) in Fig. 6.14 was a lowpass filter with bandwidth 0-3400 Hz. Applying a filter with bandwidth 0-1000 Hz to the same signal smooths out most of the kinks - trace (b) of Fig. 6.14(b). Thus, Fig. 6.14 indicates two points: the filter bandwidth used is important and secondly, the onset of the basic

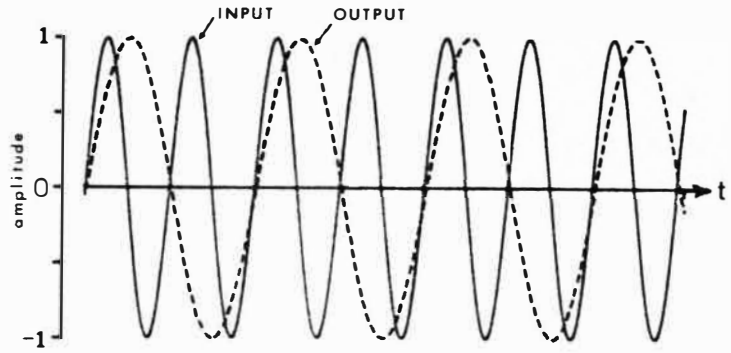


(a) unwrapping (division by 2)

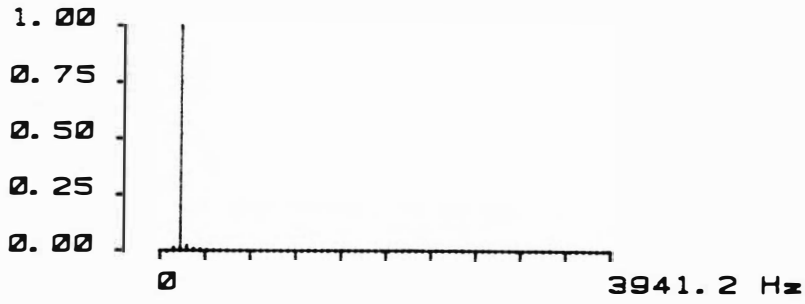


(b) wrapping (multiplication by 2)

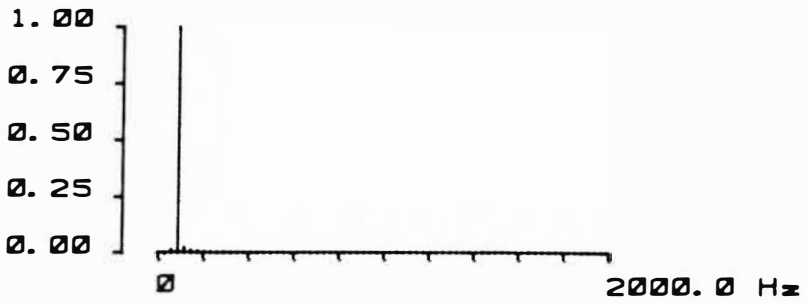
Fig. 6.11: Adaption of the basic convolution relation of Fig. 6.8 to implement the "unwrapping" and "wrapping procedures described.



(a) signal waveforms

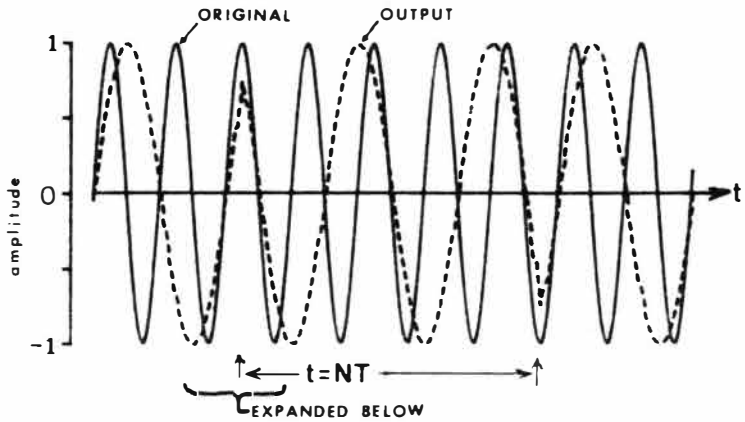


(b) input spectrum

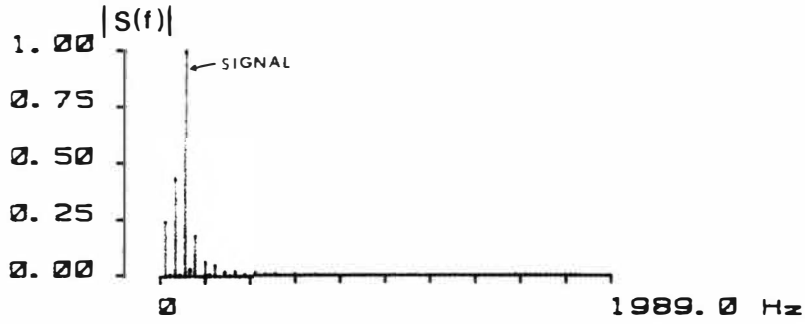


(c) output spectrum

Fig. 6.12: Phase unwrapping procedure of Fig. 6.11(a) applied to a single sinusoid whose period is an integer multiple of the filter length. Filter bandwidth=0-3400 Hz.



(a) signal waveforms



(b) spectrum of the output.

Fig. 6.13: Unwrapping procedure applied to a sinusoid whose period is not an integer multiple of the filter length. N=90, filter bandwidth= 0=3400 Hz.

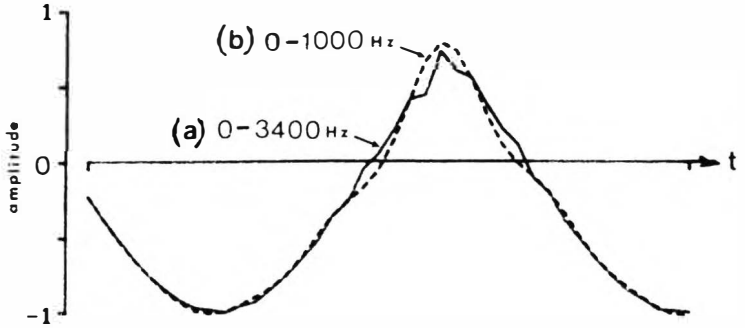


Fig. 6.14: Expansion of the discontinuity of Fig. 6.13 as a function of different filter bandwidths

problem (the discontinuity) occurs as the peak of the impulse response approaches the edge of the (shift) register.

The experiments used a filter length N=90 - ie an even length with the input data sampling rate 8000 Hz. Employing an odd length filter (eg N=91) gives no observable difference in the behaviour described above.

The procedure outlined above is termed "phase unwrapping" because of the way in which the apparent varying time delay (Fig. 6.10) appears to "unwrap" the input signal rate of change of phase.

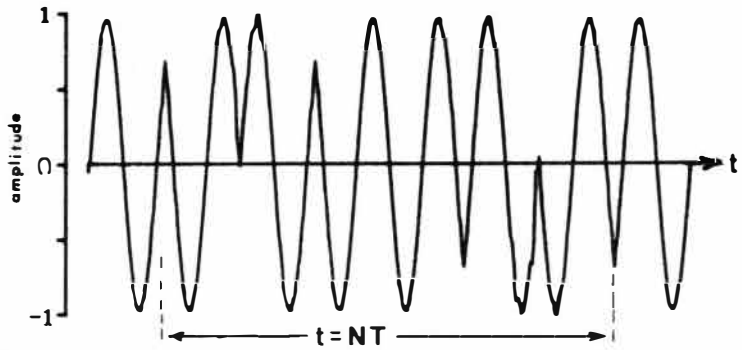
The discussion has focused on division by 2. Other, integer division factors can be achieved by varying the ratio of the clock rates shown in Fig. 6.11.

6.3.3 Frequency expansion

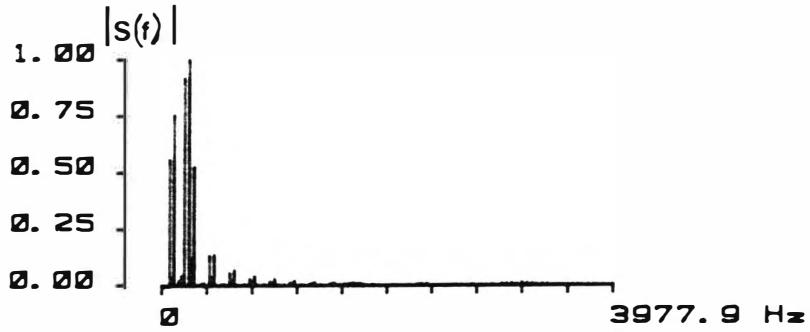
To multiply the compressed signal (sinusoid) back up to its original position, the identical filter used in the unwrapping procedure is used. However, in this case, the filter coefficients are stepped in the reverse direction to the unwrapping case - ie this time the input signal is being "wrapped up". However, because two output samples are required for every one input sample, the procedure adopted is that for each input sample, the filter coefficents are shifted (to the left in Fig. 6.11) twice, with an output sample computed at each shift. This "wrapping up" process exactly complements the initial "unwrapping process".

As for the unwrapping process, provided the signal frequency exactly matches the filter period, there are no discontinuities in the output waveform.

For a signal frequency that is not an integer multiple of the filter period, the double process of unwrapping and then wrapping does not compensate for the original discontinuties introduced in the unwrapping process. Instead, many additional artifacts (discontinuities) are introduced into the output signal. This is believed to be because the signal applied for multiplication contains harmonics introduced by the original division process. This phenomenom is shown in Fig. 6.15. Using the output shown in Fig. 6.13 as the input to the multiplication process gives the result shown in Fig. 6.15. There are now multiple discontinuities and the amplitude



(a) regenerated signal waveform. Sinusoid used is the same as that used for Fig, 6.13.



(b) spectrum of regenerated output

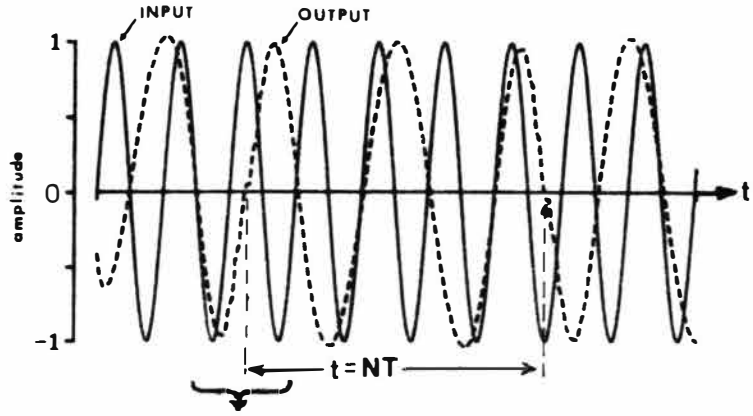
Fig. 6.15: Consecutive applications of the unwrapping and wrapping procedures for a sinusoid whose period is not an integer multiple of the filter length. Filter bandwidth=0-3400 Hz.

spectrum now has significant large amplitude extraneous harmonic components.

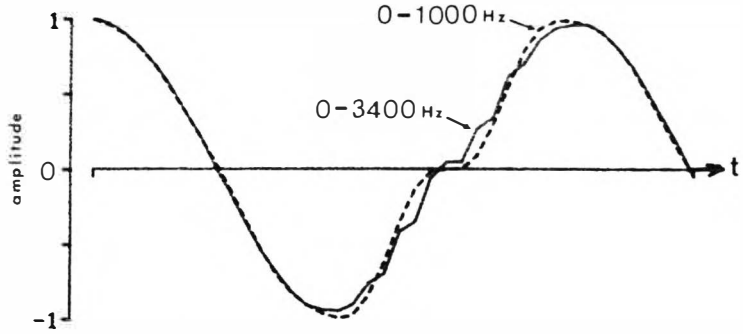
6.3.4. Eliminating the discontinuity

Fig. 6.10 suggested one way of viewing the effect of rotating the impulse response coefficients. Essentially it describes the time (delay) relationship between the input and output. The root of the problem, as already mentioned, occurs at the point where the peak of the impulse response wraps from one end of the (shift) register to the other. This corresponds to the plot in Fig. 6.10 jumping from $-\omega T(N-1)$ to zero. Invoking the time shifting property of the Fourier transform, this "jump" in the delay implies a jump, or discontinuity, in the instantaneous phase of the output signal.

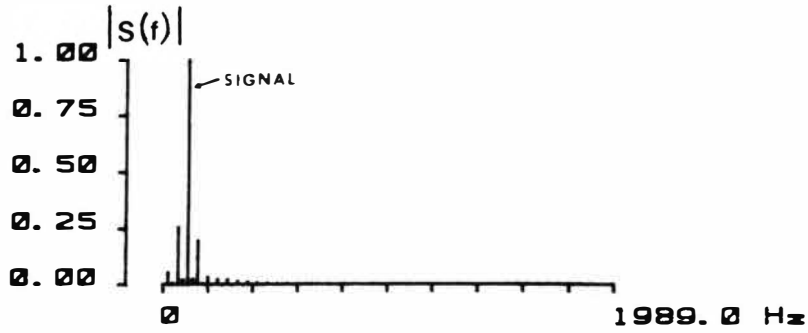
Using an identical length (N=90) Hilbert filter exactly equivalent



(a) signal waveforms



(b) expansion of the same region of the discontinuity of Fig. 6.14 vs different filter bandwidths



(c) spectrum of the output signal in comparison to that of Fig. 6.13(b).

Fig. 6.16: Hilbert version of the data in Fig. 6.13: sinusoid period not an integer multiple of the filter length. N=90; filter bandwidth= 0-3400 Hz.

in frequency response to the original filter, the Hilbert signal equivalent to Fig. 6.13 is shown in Fig. 6.16(a) with the same expansion region of the discontinuity in Fig. 6.14 shown in Fig. 6.16(b). Observe that the instant of wrap (of the filter coefficients) this time gives a single backward step in the output waveform and then continues in the same direction. Furthermore, the spectrum of this output signal - Fig. 6.16(c) compared with Fig. 6.13(b) shows the amplitudes of the extraneous harmonics are lower.

The observations above suggest a possible solution by invoking the analytic representation of equations (6.1) - (6.4) and computing instantaneous frequency or phase at the instant of the filter peak wrapping around. This value is then added to the next set of output samples as a constant until the next wrap, at which time the "constant" is updated.

Such a procedure may be satisfactory for genuinely periodic signals. However, a speech signal is only relatively pitch constant over short time intervals. Consequently, the instantaneous frequency will vary between pitch periods. This suggests that the filter length N used in the unwrapping procedure should be very short so that its time length NT is of the order of one pitch period or less.

6.3.5. Applying a speech signal

Given the reported behaviour so far, it is obvious that the procedure as described will introduce artifacts in a speech signal. However, using the original 90 point lowpass filter 0-3400 Hz, and without any form of correction attempted, the 5 second speech segment of male speaker M1 used in Chapter 5 was applied consecutively to the phase unwrapping and wrapping procedures. When played back through the D/A converter at sampling rate of 4032 Hz (half the input rate), the unwrapped (half bandwidth) speech was perceived as having a much reduced pitch , but the rate of articulation (ie the time scale) was unchanged. It was also essentially unintelligible.

Subsequent "wrapping" of the compressed speech gave a speech signal that had restored its pitch range but contained considerable interference best described as "buzzing". Based on the spectrum shown in Fig. 6.13(b), the "buzzing" in the re-generated speech is conjectured to be the harmonics created by the periodic wrap of the filter coefficients which generates the discontinuites - ie the

(frequency) spacing between the spectral lines of Fig. 6.13(b).

The regenerated speech had poorer quality and had lost intelligibilty but was nevertheless understandable. It also sounded mechanical and gave the impression of talking in a monotone - much of the natural inflexion of the original speech was lost.

6.4 CONCLUDING COMMENTS

The experiments reported above are very incomplete, and the explanation lacks rigour. However, the intention was simply to describe a particular, undeveloped, train of thought that gave rise to a unique set of experiments. The validity of the basic premise that applying a linear phase filter to a signal in a particular way allows the bandwidth and harmonic structure of a signal to be modified is justified by the simple experiment using speech. The fundamental question that remains to be answered, is whether the artifacts introduced by processing can either be eliminated, or reduced to a sufficiently low level that they are not perceivable.

CHAPTER 7

CONCLUSIONS AND FUTURE WORK

7.1 GENERAL CONCLUSIONS

The stated intention of this thesis was to investigate, via computer simulation in non real time, issues related to the manipulation of speech phase by the process of linear filtering. An additional, implied goal, was that the knowledge gained by the investigation would find application in the field of reduced bit-rate digital speech coding.

To achieve the stated goals, this work has applied knowledge and techniques of three distinct fields of endeavour - linear and non-linear phase FIR digital filters; the speech signal, and digital speech coding. In so doing, a contribution to each field has been made.

This work reports three principal investigations. The first is on FIR digital filters. The emphasis, and desired goal, in this study on digital filters was to develop a computer program that could generate the coefficients of both linear and non-linear phase FIR digital filters. In the course of this investigation, five representative algorithms were implemented for linear phase FIR filters. The study focused on practical issues related to their use. The study compares the performance of each algorithm based on a defined common comparison base. The focus of the comparison is on the end result achieved, rather than the more usual practice found in the literature that compares relative algorithmic efficiency.

During the course of this study, Algazi's iterative procedure is examined from the point of view of its use directly, rather than the original report that suggested it was useful as a "finishing" technique for another design algorithm. The study also points out some problems with Hilbert filters designed using McLellan's algorithm. These problems were not anticipated and do not appear to have been reported in the literature.

Chapter 3 investigated the issues involved in generating a non-linear phase filter. Here, three important results are achieved. First, an examination of the required impulse response length showed that in principle, longer impulse response lengths are required in the

presence of a non-linear phase. At the same time, it was also demonstrated that the desired phase response was a significant influence on the amount of extension actually required.

Secondly, this chapter considers an extension of Algazi's iterative procedure to the non-linear phase case. It was demonstrated that this method was very suitable and offered major advantages in speed and simplicity over traditional methods such as Cuthbert's. The one sacrifice was that, in general, Algazi's method had a poorer stopband performance than virtually every other algorithm used.

The third important result stems from the discussion on the specific random phase filters required for the work on phase randomisation of speech. This is that clear demonstration of the effect of a highly non-linear phase involving discontinuities has on the required impulse response length and the resulting magnitude response. Such a result might be considered trivial because it can be easily justified using the duality of the Fourier transform. However, the result is not necessarily intuitively obvious to those either new to the subject, or unpracticed in the application of Fourier theory.

An indirect outcome of the investigation on FIR filters is the development of an integrated software package that implements five linear phase design algorithms and two non-linear phase design algorithms.

The second principal investigation is reported in chapter 5. The interest here is first, to see if a worthwile reduction in Peak Factor can be achieved more easily. Secondly, what are the differences - both objective and acoustical- in the phase randomised speech, both between original and randomised speech and male versus female. This study gives four important direct results. First, the FIR filter method is feasible and its performance, at least for one word studied, is better than David's original method because peak factor reduction is achieved in utterances that are normally considered as unvoiced and therefore lacking a periodic structure.

Secondly, the study postulates, in the absence of any known method of analytic solution, an irreducible minimum peak factor that will, in practice never be achieved by a real signal composed of more than one harmonic. At the same time it is shown that the peak factor reduction is consistent for both males and females with relatively modest

reductions for male speech but much less for female speech.

Thirdly, the analytic zero description of the speech signal allows an unusual insight into characteristics of speech that otherwise lack an adequate means of explanation. In particular, an explanation is afforded as to why female speech is widely regarded as sounding "smoother". The work here suggests that the smoothness of female speech is due to a naturally more even distribution of analytic zeros. This more even distribution translates to a naturally more random relationship between the harmonic phases.

Finally, phase randomisation of speech is shown to have a marked effect on the probability density function of male speech but not female speech. For male speech, a noticeable asymmetry in the original is largely eliminated by the phase randomisation. In contrast, the distribution for female speech is naturally symmetrical. Thus, the phase randomisation will in some cases only make matters worse by upsetting this natural symmetry. This different behaviour helps to explain the difference in peak factor reduction between males and females.

The third and final investigation and experimental work is documented in Chapter 6. This investigation relates to methods of achieving bit-rate reduction in speech coding. A radically different approach to the basic requirement of harmonic bandwidth compression and expansion of the speech signal by the application of a linear phase FIR filter is described. Although the experiments are incomplete, using single sinusoids, and one example of speech, the basic idea and method was shown to be valid. One of the attractive aspects of this method is that, with the single exception of Malah's time domain harmonic scaling approach, it offers considerable savings, both computationally and practically, over any other method of frequency division and expansion described in the literature.

7.2 FUTURE WORK

In chapter 2, the reported problem of McLellan's algorithm when designing Hilbert filters requires investigation. Such an investigation would be largley mathematical in nature. The principal questions are (a): is the problem a software one (coding errors of the algorithm) or (b): is the problem fundamental to the algorithm and its underlying mathematical basis?.

Also in chapter 2 and again largely mathematical in nature, the original work of Algazi and Suk needs to be re-assessed carefully with respect to the iterative algorithm that they develop. This is because the implication of the original work was that the traditionally difficult problem of extracting progressively higher order prolate spheroidal sequences could be avoided, but still provide the same optimal solution. In practice, the iterative algorithm never achieved the level of performance implied. In particular, filters designed using the iterative procedure consistently gave poorer stopband performance than any other algorithm tried.

In chapter 3, Steiglitz's method for designing non-linear phase FIR filters was described but not implemented. This was simply because the original report simply outlined the principle of the method and gave no useful mathematical detail. In addition, he noted that a specially coded version of the simplex algorithm was used. Thus, any other reader wishing to use the basic approach must "re-invent" the underlying mathematical detail that constitutes the design algorithm. As such, the problem is largely mathematical in nature and requires a familiarity with the theory and application of the simplex optimisation algorithm.

The work on phase randomisation of speech suggested that the lowest peak factor would be achieved when the zeros of the speech (signal) were all real (ie zero crossings). Clear evidence of a certain amount of complex to real zero conversion was demonstrated. However, the basic assumption requires validation. Also required is more detailed investigation of the basic analytic zero distribution of male and female speech before and after attempted phase randomisation. This would help in two ways: to account for the differences in peak factor reduction between males and females and help point the way to achieving a better peak factor reduction.

Finally, the new approach to achieving bandwidth control of a periodic signal described in chapter 6 is wide open. The fundamental problem to be overcome has been identified and a possible means of solution proposed. In addition, consideration must be given to two distinct types of excitation that underly speech: the voiced segments are basically periodic. However, the unvoiced segments are noise-like. What is the behaviour of the (un)wrapping procedure with a noise source applied?

REFERENCES

ALGAZI V.R., MINSOO SUK, (1975) On the Frequency Weighted Least-Square Design of Finite Duration Filters., IEEE Trans. on Circuits and Systems, Vol CAS-22 No 12 December 1975 pp 943-953.

ALHBORG H., (1978) Speech Levels in the Swedish Telephone Network. TELE (English Edition) Vol lXXX No1/1978 pp 22-26.

ALLEN J.B., RABINER L.R., (1977) A Unified Approach to Short-Time Fourier Analysis and Synthesis. Proc IEEE Vol 65 No 11 Nov 1977 pp 1558-1564

ANTONIOU A., (1983) New Improved Method for the Design of Weighted-Chebyshev, Norecursive Digital Filters., IEEE Trans on Circuits and Syst. Vol CAS-30 No 10 October 1983 pp 740-750.

ATAL B.S., HANAUER S.L., (1971) Speech Analysis and Synthesis by Linear Prediction of the Speech Wave. J. Acoust. Soc. Am. Vol 50 No 2 1971 pp 637-655

BAR-DAVID I., (1974) An Implicit Sampling Theorem for Bounded Bandlimited Functions., Inform. Contr. Vol 24 Jan 1974 pp 36-44

BERGLAND G.D., DOLAN M.T., Subroutine FFT842 in Programs for Digital Signal Processing, IEEE Press, 1979, pp 1.2-1

BOGERT B.P., (1956) The Vobanc - A Two-to-one SpeechBand-width Reduction System. J. Acoust. Soc. Am. Vol 28 No 3 May 1956 pp 399-404

BOGNER R.E., (1965) Frequency Division in Speech Bandwidth Reduction. IEEE Trans on Commun. Tech. Vol COM-13 No 4 December 1965 pp 438-451

BOGNER R.E., FLANAGAN J.L. (1969) Frequency Multiplication of Speech Signals. IEEE Trans. on Audio and Acoust. Vol AU-17 No 3 September 1969 pp 202-208

BOND F.E., CAHN C.R., (1958) On sampling the Zeros of Bandwidth Limited Signals., IRE Trans Vol IT-4 No 3 September 1958 pp 110-113

BOZIC S.M., (1979a) Digital and Kalman Filtering,

BOZIC S.M., (1979b) Digital and Kalman Filtering, Ch 7 pp 92-94.

BRADY PT (1965) A Statistical Basis for Objective Measurement of Speech Levels. Bell Sys. Tech. J. Vol 44 1965 pp 1453-1486 (September)

BRADY PT (1968) Equivalent Peak Level: A Threshold-Independent Speech-Level Measure. J. Acoust. Soc. Am. Vol 44 No 3 1968 pp 695-699 (March)

BRIGHAM E.O., (1974) The Fast Fourier Transform., Prentice Hall, 1974.

BROPHY F.J., SALAZAR A.C. (1975) Two Design Techniques for Digital Phase Networks., BSTJ Vol 54 No 4 April 1975 pp 767-781

CAMPBELL L.S., (1980) An Automatic Signal Level Meter for Telecommunications Applications., Christchurch, University of Canterbury 1972, 132p. (Thesis: M.E. : Electrical)

CHING P.C., (1983) A Study of Time Encoded Speech. Radio and Electronic Engineer, Vol 53 No 9 September 1983 pp 321-324

CHINN H.A., GANNETT D.K., MORRIS R.M., (1940) A New Standard Volume Indicator and Reference Level. Bell Sys. Tech. J. Vol 19 No 1 1940, pp 94-137 (January)

CLERGEOT H., SCHARF L.L., (1978) Connections Between Classical and Statistical Methods of FIR Digital Filter Design., IEEE Trans on Acoust. Speech and Sig. Proc. Vol ASSP-26 No 5 October 1978 pp 463-465.

COOLEY J.W, TUKEY, J.W., (1965) An algorithm for the machine calculation of complex Fourier series. Mathematics of Computation, Vol 19 No 90 1965 pp 297-301

COX R.V., CROCHIERE R.E., JOHNSTON J.D. (1983) Real Time Implementation of Time Domain Harmonic Scaling of Speech for Rate Modification and Coding. IEEE Trans. Acoust. Speech and Sig. Proc. Vol ASSP-31 No 1 Feb 1983 pp 258-271

CROCHIERE R.E., WEBBER S.A., FLANAGAN J.L. (1976) Digital Coding of Speech in Sub-bands. Bell Sys. Tech. J. Vol 55 No 8 October 1976 pp 1069-1085

CROCHIERE R.E. (1977) On the Design of Sub-band Coders for Low-Bit-Rate Speech Communication. Bell Sys. Tech. J. Vol 56 No 5 May-June 1977 pp 747-770

CROCHIERE R.E. (1981) Sub-Band Coding. Bell Sys.Tech. J. Vol 60 No 7 September 1981 pp 1633-1653

CROCHIERE R.E., FLANAGAN J.L. (1983a) Current Perspectives in Digital Speech. IEEE Communications Magazine Jan 1983 32-40

CROCHIERE R.E., RABINER L.R. (1983b) Multirate Digital Signal Processing. Prentice-Hall, 1983

CUTHBERT L.G., (1974a) Nonrecursive Digital Filters with Crescent Ripple., Electron. Lett Vol 10 NO 19 pp 397-399. (19 September 1974)

CUTHBERT L.G. (1974b) Optimising Non-Recursive Digital Filters to Non-linear Phase Characteristics., Radio Electron. Eng. Vol 44 No 12 December 1974 pp 645-651.

DAGUET J.L. (1963) Speech Compression CODIMEX System. IEEE Trans. on Audio and Electroacoust. Vol AU-11 March-April 1963 pp 63-71

DAVENPORT W.B. (1952) An Experimental Study of Speech-Wave Probability Distributions. Acoust. Soc. Am. Vol 24 No 4 July 1952 390-399

DAVID E.E., MILLER, J.E., MATHEWS M.V. (1961) Monaural Phase Effects in Speech Perception., Proc. 3rd Int. Congress on Acoustics, 1959, Stuttgart, L. Cramer, Ed. (Elsevier Publ. Co., Amsterdam 1961) Vol 1 pp 227-229

EBERT S., HEUTE U. (1983) Accelerated Design of Linear or Minimum Phase FIR Filters with a Chebyshev Magnitude Response., Proceedings IEE Vol 130 Pt G No 6 December 1983 pp 267-270.

FARDEN D.C., SCHARF L.L.,(1974) Statistical Design of Nonrecursive Digital Filters., IEEE Trans on Acoust. Speech and Sig. Proc. Vol ASSP-22 No 3 June 1974 pp 188-200.

FARDEN D.C., SCHARF L.L.,(1975) Author's replay to Butler's Comments on "Statistical design of nonrecursive digital filters," IEEE Trans. on Acoust. Speech and Sig. Proc. Vol ASSP-23 pp 495-497, 1975.

FLANAGAN J.L. Speech Analysis, Synthesis and Perception., 2nd Edition, Springer-Verlag, 1972

FLANAGAN J.L., GOLDEN R.M. (1966) Phase Vocoder. Bell Sys. Tech. J., Nov 1966 pp 1493-1509.

FLANAGAN J.L., SCHROEDER M.R., ATAL B., CROCHIERE R.E. (1979) Speech Coding. IEEE Trans on Comm. Vol COM-27 No 4 April 1979 pp 710-736

FLANAGAN J.L. (1980a) Parametric Coding of Speech Spectra. J. Acoust. Soc. Am. Vol 68(2) Aug 1980 pp 412-419

FLANAGAN J.L., CHRISTENSEN S.W. (1980b) Computer Studies on Parametric Coding of Speech Spectra. J. Acoust. Soc. Am. Vol 68(2) Aug 1980 pp 420-430

FLANAGAN J.L., CHRISTENSEN S.W. (1980c) Technique for Frequency Division/Multiplication of Speech Signals.J. Acoust. Soc. Am. Vol 68(4) Oct 1980 pp 1061-1068

FLETCHER R., POWELL (1963), A Rapidly Convergent Descent Method for Minimisation

FLETCHER R. (1971) A Modified Marquardt Subroutine for Non-linear Least Squares, UKAEA Research Group Report AERE-R 6799, 1971.

GOLD B., (1964) Experiment with Speechlike Phase in a Spectrally Flattened Pitch-Excited Channel Vocoder., J. Acoust. Soc. Am. Vol 36 No 10 October 1964 pp 1892-1894

GOLD B., JORDAN K.L., (1969a) A Direct Search Procedure for Designing Finite Duration Impulse Response Filters., IEEE Trans. Audio and Electroacoust. Vol AU-17 No 1 March 1969 pp 33-36.

GOLD B., RABINER L.R. (1969b) Parallel Processing Techniques for Estimating Pitch Periods of Speech in the Time Domain., J. Acoust. Soc. Am. Vol 46 No 2 Pt 2 pp 442-448 August 1969

GOLDBERG E, KURSHAN R, MALAH D (1981) Design of Finite Impulse Response Filters with Nonlinear Phase Response., IEEE Trans. on Acoustics, Speech and Sig. Processing, Vol ASSP-29 No 5 October 1981 pp 1003-1010

GOLDEN R.M. (1968) Vocoder Filter Design: Practiccal Considerations. J. Acoust. Soc. Am. Vol 43 No 3 April 1968 pp 803-810

GREGORIAN R.G., TEMES G.C. (1978) Design Techniques for Digital and Analog All-Pass Circuits., IEEE Trans. on Circuits and Systems, Vol CAS-25 No 12 Decemeber 1978 pp 981-988.

HAMILTON A.C.J., (1985) Parameters of the Analytic Vector Representation of Speech, 1985, Christchurch, University of Canterbury. (Thesis: PhD: Electrical)

HARRIS F.J., (1978) On the Use of Windows for Harmonic Analysis with the Discrete Fourier Transform. Proceedings of IEEE Vol 66 No 1 January 1978 pp 51-83.

HELMS H.D., (1968) Nonrecursive Digital Filters: Design Methods for Achieving Specifications on Frequency Response.,IEEE Trans. on Audio and Electroacoust. Vol AU-16 No 3 September 1968 336-342.

HELMS H.D., (1971) Digital filters with Equiriplle or Minimax Responses., IEEE Trans. on Audio and Electroacoustics Vol AU-19 No 1 March 1971 pp 87-93.

HERRMANN O., (1970a) Design of Nonrecurisive Digital Filters with Linear Phase., Electron. Lett. Vol 6 No 11 28 May 1970 pp 328-329

HERRMANN O., (1970b) On the Approximation Problem in Nonrecursive Digital Filter Design., IEEE Trans. on Circuit Theory, Vol CT-18 No 3 May 1971 pp 441-413.

HERRMAN O, SCHUESSLER W, (1970c) Design of Nonrecursive Digital Filters with Minimum Phase., Electron. Lett. Vol 6 No 11 28 May 1970 pp 329-330.

HERMANN O., RABINER L.R., CHAN D.S.K., (1973) Practical Design Rules for Optimum Finite Impulse Response Low-Pass Digital Filters. BSTJ Vol 52 No 6 July-August 1973 pp 769-799.

HILLE E.,(1959) Analytic Function Theory Vol I, Ginn and Co. 1959, pp 207

HOLT A.G.J., ATTIKIOUZEL J, BENNET R (1976) Iterative Technique for Designing Non-Recursive Digital Filter non-linear Phase Characteristic., Radio Electron. Eng. Vol 46 No 12 December 1976 pp 589-592.

IEEE Programs for Digital Signal Processing, IEEE Press, 1979

JAYANT N.S., NOLL P., (1984) Digital Coding of Waveforms. Prentice-Hall, 1984

JENKINS M.A., TRAUB J.F.,(1972) Algorithm 419: Zeros of a Complex Polynomial. Communications of ACM Vol 15 No 2 pp 97-99 February 1972.
[see also Remark by D.H. Withers, Ibid, Vol 17 No 3 p 157 (March 1974)]

KAISER J.F., (1966) Digital Filters, Ch. 7 in "Systems Analysis by Digital Computer", Kuo and Kaiser, Wiley 1966 pp 228-243.

KAISER J.F, (1974) Nonrecursive Digital Filter Design Using the Io-Sinh Window Function., Proc IEEE Int. Symp. on Circuits and Syst., Apr 22-25 1974 pp 20-23

KALOUPTSIDIS N., KOYAS G.D., Efficient Block LS Design of FIR Filters with Linear Phase., IEEE Trans on Acoust. Speech and Sig. Proc., Vol ASSP-33 No 6 Decemeber 1985 pp 1435-1444.

KAMP Y., WELLEKENS C.J., Optimal Design of Minimum Phase FIR Filters. IEEE Trans. Acoust. Speech and Sig. Proc., Vol ASSP-31 No 4 August 1983 pp 922-926

KANG G.S., EVERETT S.S. (1983) Improvement of the Narrowband LPC Synthesis., IEEE Int. Conf. on Acoust., Speech and Sig. Proc. 1983 Vol 1 pp 89-82.

KELLOG W.C., (1972) Time Domain Design of Nonrecursive Least Mean-Square Digital Filters., IEEE Trans on Audio and Electroacoust. Vol AU-20 No 2 June 1972 pp 155-158.

KING R.A., GOSLING W., (1978) Time Encoded Speech. Elec. Lett. 20 July 1978 Vol 14No 15 pp 456-457

LEVINSON N. (1947) The Wiener RMS Error Criterion in Filter Design and Prediction., Appendix B in "Extrapolation, Interpolation and Smoothing of Stationary Time Series by N. Wiener", MIT Press 1964.

LEWIS J.T., MURPHY R., TUFTS D.W. Design of Minimum Noise Digital Filters Subject to Inequality Constraints Using Quadratic Programming., IEEE Trans on Acoust. Speech and Sig. Proc. Vol ASSP-24 October 1976 pp 434-436.

LIM J.S., OPPENHEIM A.V. (1979) Enhancement and Bandwidth Compression of Noisy Speech. Proc IEEE Vol 67 No 12 December 1979 1586-1604

MALAH D., (1979) Time Domain Algorithms for Harmonic Bandwidth Reduction and Time Scaling of Speech Signals. IEEE Trans. on Acoust. Speech and Sig. Proc. Vol ASSP-27 No 2 April 1979 pp 121-133

MALAH D., FLANAGAN J.L. (1981) Frequency Scaling of Speech Signals by Transform Techniques. Bell Sys. Tech. J. Vol 60 No 9 Nov 1981 pp 2107-2156

MANOLAKIS D., KALOUPTSIDIS N., CARAYANNIS G. (1982) Efficient Determination of FIR Wiener Filters with Linear Phase., Electron. Lett. Vol 18 No 10 13 May 1982 pp 429-431.

MARCOU P., DAGUET J. (1955) New Methods of Speech Transmission. Proceedings of 3rd Symposium on Information Theory, London 1955

MARKEL J.D., GRAY A.H, (1976) Linear Prediction of Speech. Springer-Verlag, N.Y., 1976

MARPLE L.S., (1981) Efficient Least Squares FIR System Identification. IEEE Trans on Acoust. Speech and Sig. Proc. Vol ASSP-29 No 1 February 1981 pp 62-73.

MATHES R.C., MILLER R.L. (1947) Phase Effects in Monaural Perception. J. Acoust. Soc. Am. Vol 19 No 5 September 1947 pp 780-784

MATHEWS J.D., BREAKALL J.K., KARAWAS G.K., (1985) The Discrete Prolate Spheroidal Filter as a Digital Signal Processing Tool. IEEE Trans on Acoust. Speech and Sig. Proc, Vol ASSP-33 No 6 December 1985 pp 1471-1478.

McCLELLAN J.H., PARKS T.W., (1973a) A Unified Approach to the Design of Optimum FIR Linear-Phase Digital Filters. IEEE Trans. on Circuit Thoery Vol CT-20 No 6 December 1973 pp 697-701.

McCLELLAN J.H., PARKS T.W., RABINER L.R., (1973b) A Computer Progam for Designing Optimum FIR Linear Phase Digital Filters. IEEE Trans. on Audio and Electroacoustics Vol AU-21 No 6 December 1973 pp 506-525.

McCREARY T.J., (1972) On Frequency Sampling Filters. IEEE Trans. on Audio and Electroacoust. Vol AU-20 No 3 August 1972 pp 222-223.

MORRIS L.R., (1972) The Role of Zero Crossings in Speech Recognition and Analysis., 1972 Proc Conf. on Speech Communication and Processing April 1972 paper L7 pp 446-450

NIEDERJOHN R.J, HAWORTH .G. (1983) The Relationship between the rms level and the average absolute magnitude of long-time continuous speech. J. Acoust. Soc. Am. Vol 74 No 2 pp 444-446 1983 (August)

OPPENHEIM A.V., SCHAFER R.W., (1975a) Digital Signal Processing, Prentice Hall, 1975 Ch. 4 pp 136-165.

OPPENHEIM A.V., SCHAFER R.W., (1975b) Digital Signal Processing, Prentice Hall, 1975, section 7.2 pp 345-353.

PAEZ M.D., GLISSON T.H. (1972) Minimum Mean_Squared Error Quantisation in Speech PCM and DPCM Systems.IEEE Trans. on Communications Vol COM-20 No April 1972 pp 225-230

PAPOULIS A., (1962a) The Fourier Integral and its Applications, McGraw-Hill 1962.

_____, (1962b) ibid, Chapter 6 pp 114-115.

PAPOULIS A., (1967) Limits on Bandlimited Signals., Proceedings of IEEE Vol 55 No 10 October 1967 pp 1677-1686.

PAPOULIS A., BERTRAN M.S., (1972) Digital Filtering and Prolate Functions IEEE Trans on Circuit Theory Vol CT-19 No 6 November 1972 pp 674-681.

PARKS T.W., McCLELLAN J.H., (1972) Chebyshev Approximation for Nonrecursive Digital Filters with Linear Phase. IEEE Trans on Circuit Theory Vol CT-19 No 2 March 1972 pp 189-195.

PIWNICKI K., (1983) Modulation Methods Related to Sine-Wave Crossings. IEEE Trans on Comm. Vol COM-31 No 4 April 1983 pp 503-508

PLOMP R., STEENEKEN H.J.M. (1969) Effect of Phase on the Timbre of Complex Tones. J. Acoust. Soc. Am. Vol 46 No 2 pt 2 pp 409-421 1969

PORTNOFF M.R., (1976) Implementation of the Digital Phase Vocoder Using the Fast Fourier Transform. IEEE Trans. Acoust. Speech and Sig. Proc. Vol ASSP-24 No 3 June 1976 pp 243-248

PURTON R.F. (1962) A Survey of Telephone Speech-Signal Statistics and their Significance in the choice of a P.C.M Companding Law. PROC IEE Vol 109B 1962 pp 60-66

RABINER L.R., GOLD B., McGONEGAL C.A., (1970) An Approach to the Approximation Problem for Nonrecursive Digital Filters. IEEE Trans. on Audio and Electroacoust. Vol AU-18 No 2 June 1970 pp 83-106.

RABINER L.R., (1971) Techniques for Designing Finite-Duration Impulse-Response Digital Filters., IEEE Trans on Communications Vol COM-19 No 2 April 1971 pp 188-195.

RABINER L.R., (1972) Linear Program Design of Finite Impulse Response (FIR) Digital Filters. IEEE Trans on Audio and Electroacoust. Vol AU-20 No 4 October 1972 pp 280-288.

RABINER L.R., (1973) Approximate Design relationships for Low-Pass FIR Digital Filters. IEEE Trans on Audio and Electroacoust. Vol AU-21 No 5 October 1973 pp 456-461

RABINER L.R., GOLD B., (1975a) Theory and Application of Digital Signal Processing, Prentice Hall 1975, Chapt. 2.

RABINER L.R., McCLELLAN J.H., PARKS T.W., (1975b) FIR Digital Filter Design Techniques Using Weighted Chebyshev Approximation. Proc. IEEE Vol 63 No 4 April 1975 pp 595-610.

RABINER L.R., SCHAFER R.W., (1976) A Comparative Performance Study of Several Pitch Detection Algorithms., IEEE Trans. Acoust. Speech and Sig. Proc., Vol ASSP-24 No 5 October 1976 pp 399-418.

RABINER L.R., SCHAFER R.W. (1978) Digital Processing of Speech Signals, Prentice Hall, 1978

RADER C.M., GOLD B., (1967) Digital Filter Design Techniques in the Frequency Domain, Proc. IEEE Vol 55 No 2 February 1967 pp 149-171.

REQUICHA A.A.G., VOELCKER H.B., (1970) Design of Nonrecursive Filters by Specification of Frequency-Domain Zeroes. IEEE Trans. on Audio and ELectroacoust. Vol AU-18 No 4 December 1970 pp 465-470.

REQUICHA A.A.G., (1980) The Zeroes of Entire Functions: Theory and Engineering Applications. Proc IEEE Vol 68 No 3 March 1980 pp 308-328

RICE J.R., (1964a) The Approximation of Functions, Addison-Wesley 1964 Vol 1 Chpt. 3 pp 52-93

RICE J.R., (1964b) ibid, Vol 1 pp 171-180

RICE J.R., (1964c) ibid, Vol 1 pp 30-34

ROSS M.J., SHAFFER H.L., COHEN A., FREUDBERG R., MANLEY H.J. (1974) Average Magnitude Difference Function Pitch Extractor. IEEE Trans. Acoust. Speech and Sig. Proc., Vol ASSP-22 No 5 October 1974 pp 353-362

SCHAFER R.W., RABINER L.R. (1970) System for Automatic Formant Analysis of Voiced Speech., J. Acoust. Soc. Am. Vol 47 No 2 February 1970 pp 634-648

SCHAFER R.W., RABINER L.R. (1973) Design and Simulation of a Speech Analysis-Synthesis System Based on Short Time Fourier Analysis. IEEE Trans. Audio and Electroacoust. Vol AU-21 No 3 June 1973 pp 165-174

SCHAFER R.W. (1974) FIR Digital Filter Banks for Speech Analysis. Bell Sys. Tech. J. Vol 54 No 3 March 1975 531-544

SCHROEDER M.R. (1966) Vocoders., Proc IEEE Vol 54 No 5 pp 720-734 May 1966.

SCHROEDER M.R., FLANAGAN J.L., LUNDRY E.A., (1967) Bandwidth Compression of Speech by Analytic-Signal Rooting. Proc. IEEE Vol 55 No 3 March 1967 pp 396-401

SCHROEDER M.R. (1975) Models of Hearing., Proc IEEE Vol 63 No 9 Sept 1975 1332-1350

SEKEY A., (1970) A Computer Simulation Study of Real-Zero Interpolation. IEEE Trans Audio and Electroacoust. Vol AU-18 No 1 March 1970 pp 43-54

SENEFF S. (1982) System to Independently Modify EXcitation and/or Spectrum of Speech Waveform Without Explicit Pitch Extraction. IEEE Trans. on Acoust. Speech and Sig. Proc. Vol ASSP-30 No 4 August 1982 pp 566-578

SIEGEL L.J. (1979) A Procedure for Using Pattern Classification Techniques to obtain a Voiced/Unvoiced Classifier., IEEE Trans. Acoust. Speech and Sig. Proc., Vol ASSP-27 No 1 February 1979 pp 83-89

SIEGEL L.J., BESSEY A.C. (1982) Voiced/Unvoiced/Mixed Excitation Classification of Speech. IEEE Trans. ACoust. Speech and Sig. Proc. Vol ASSP-30 No 3 June 1982 pp 451-460

SINGLETON R.C.(1979) Mixed Radix Fast Fourier Transforms. IEEE Programs for Digital Signal Processing, IEEE Press 1979, pp 1.4.1-1.4.

SLEPIAN D., POLLACK H.O., LANDAU H.J., (1961) Prolate Spheroidal Wave Functions, Fourier Analysis and Uncertainty. BSTJ Vol 40 1961, Parts I and II pp 43-84

SLEPIAN D., (1976) On Bandwidth. Proc IEEE Vol 64 No 3 March 1976 pp 292-300

STEELE R., (1975) Delta Modulation Systems, Halsted Press, NY, 1975

STEIGLITZ K., (1979) Optimimum Design of FIR Digital Filters with Monotone Passband Response., IEEE Trans on Acoust. Speech and Sig. Proc. Vol ASSP-27 No 6 December 1979 pp 643-649.

TEMES G.C., BARCILON V., MARSHALL F.C., (1973) The Optimisation of Bandlimited Systems. Proc IEEE Vol 61 No 2 February 1973 pp 196-234

TUFTS D.W., FRANCIS J.T., (1970) Designing Digital Low-Pass Filters - Comparison of Some Methodes and Criteria., IEEE Trans on Audio and Electroacoust. Vol AU-18 No 4 December 1970 pp 487-494.

VAIDYANATHAN P.P., (195) Optimal Design of Linear-Phase FIR Digital Filters with Very Flat Passbands and Equripple Stopbands., IEEE Trans on Circuits and Syst. Vol CAS-32 No 9 September 1985 pp 904-917.

VIVALDA E., BAINCO C., MATTEONI M., (1984) Performance Evaluation of 112 Voices of a Pitch Tracking Algorithm by Comparison with a Glottal Signal Pitch Distribution Statistical Analysis. Olivetti Res. & Tech. Rev. No 2 1984 pp 73-88

VOELCKER HB (1966a) Toward a Unified theory of Modulation Part I: Phase Envelope Relationships. PROC IEEE Vol 54 No 3 pp 340-353 (march 1966)

VOELCKER HB (1966b) Toward a Unified theory of Modulation Part II: Zero Manipulation. Ibid, No 5 pp 735-755 (may 1966)

WANG D.L., LIM J.S. (1982) The Unimportance of Phase in Speech Enhancement., IEEE Trans. Acoust. Speech and Sig. Proc., Vol ASSP-30 No 4 August 1982 679-681

WOODWARD P.M., (1953) Probability and Information Theory with Applications to Radar, Pergamon Press, 1953

ZELINSKI R., NOLL P., (1977) Adaptive Transform Coding of Speech Signals. IEEE Trans. Acoust. Speech and Sig. Proc. Vol ASSP-25 No 4 August 1977 pp 298-309

ZELINSKI R., NOLL P., (1979) Approaches to Adaptive Transform Speech Coding at Low Bit Rates. IEEE Trans. Acoust. Speech and Sig. Proc. Vol ASSP-27 No 1 February 1979 pp 89-95

APPENDIX 1

THE COMPUTATIONAL ASPECTS RELATED TO IMPLEMENTING CUTHBERT'S PROCEDURE.

The DFT of the impulse response h(n) is

$$H(\omega) = \sum_{n=0}^{N-1} h(n)e^{-j\omega Tn} \qquad \text{A1.1}$$

If the impulse response is represented by two symmetric sets a(n) and b(n) that possess even and odd symmetry respectively, then:

I: For N EVEN:

$$\begin{aligned} h(n) &= a(n) + b(n), \\ h(N-1-n) &= a(n) - b(n), \end{aligned} \qquad n = 0,\ldots,(N/2)-1 \qquad \text{A1.2}$$

Substituting A1.2 in A1.1 gives

$$\begin{aligned} H(\omega) &= \sum_{n=0}^{N/2-1} [a(n)+b(n)]e^{-j\omega Tn} + \sum_{n=0}^{N/2-1} [a(n)-b(n)]e^{-j\omega T(N-1-n)} \\ &= \sum_{n=0}^{N/2-1} a(n)[e^{-j\omega Tn} + e^{-j\omega T(N-1-n)}] \\ &\quad + \sum_{n=0}^{N/2-1} b(n)[e^{-j\omega Tn} + e^{-j\omega T(N-1-n)}] \end{aligned} \qquad \text{A1.3}$$

Noting that

$$e^{-j\omega Tn} + e^{-j\omega T(N-1-n)} = e^{-j\omega T(N-1)/2}\left(e^{-j\omega T[n-(N-1)/2]} + e^{-j\omega T[n-(N-1)/2]}\right)$$

and similarly for the second term, then A1.3 becomes

$$\begin{aligned} H(\omega) = e^{-j\omega T(N-1)/2} \Bigg\{ &\sum_{n=0}^{N/2-1} 2a(n)\cos\left(\omega T[n-(N-1)/2]\right) \\ &- j \sum_{n=0}^{N/2-1} 2b(n)\sin\left(\omega T[n-(N-1)/2]\right) \Bigg\} \end{aligned} \qquad \text{A1.1}$$

Define the desired frequency response as

$$D(\omega) = R(\omega) + jQ(\omega) \qquad \text{A1.5}$$

then use an optimisation procedure to minimise the objective function

$$S = \sum_{k=0}^{M} r^2(k) = \sum_{k=0}^{M} [D(\omega)-H(\omega)]^2 W^2(\omega) \qquad \text{A1.6}$$

where $W(\omega)$ is a non-negative, even weighting function. Then, using A1.4, A1.5 in A1.6, the procedure is to separately optimise

$$S_R = \sum_{k=0}^{M} \left\{ \sum_{n=0}^{N/2-1} 2a(n)\cos(\omega_k T[n-(N-1)/2]) - R(\omega) \right\}^2 W^2(\omega) \qquad \text{A1.7a}$$

$$S_Q = \sum_{k=0}^{M} \left\{ \sum_{n=0}^{N/2-1} -2b(n)\sin(\omega_k T[n-(N-1)/2]) - Q(\omega) \right\}^2 W^2(\omega) \qquad \text{A1.7b}$$

The index k corresponds to the k'th frequency at which the error is computed. k=0 implies the DC frequency $\omega=0$. The frequency variable ω_k is in practice discrete. Given the symmetry conditions of the sequences a(n) and b(n) and consequently their frequency responses $H_a(\omega)$ and $H_b(\omega)$, only the frequencies in the first half nyquist interval $0<\omega<2\pi Fs/2$ need to be considered, where Fs is the sampling frequency of the system. Thus the M+1 frequency points only need to occupy this interval. The frequency interval is $\Delta f = Fs/2M$ and thus in A1.7

$$\omega_k T = \frac{2\pi}{Fs}.\frac{Fs}{2M}.k = \frac{\pi}{M}k, \qquad k = 0,\ldots,M \qquad \text{A1.8}$$

The subroutine VA07A requires the computation of the Jacobian matrix A and the gradient vector V

$$A(i,j) = \sum_{k=0}^{M} \frac{\partial r(k)}{\partial x(i)}.\frac{\partial r(k)}{\partial x(j)} \qquad i,j = 0,\ldots,(N/2)-1 \qquad \text{A1.9}$$

$$V(i) = \sum_{k=0}^{M} r(k)\frac{\partial r(k)}{\partial x(i)} \qquad i = 0,\ldots,(N/2)-1 \qquad \text{A1.10}$$

where x is either a or b from A1.7. Using A1.6, A1.7, A1.8 and recognising that $W^2(\omega_k)$ is simply a number (a constant), then

$$\frac{\partial r(k)}{\partial a(n)} = 2W^2(\omega_k)\cos(\tfrac{\pi}{M}k[n-(N-1)/2]) \qquad \text{A1.11a}$$

$$n=0,\ldots(N/2)-1$$

$$\frac{\partial r(k)}{\partial b(n)} = -2W^2(\omega_k)\sin(\tfrac{\pi}{M}k[n-(N-1)/2]) \qquad \text{A1.11b}$$

The order of the problem to be solved in A1.7, A1.9, A1.10 is

controlled by M and N and will have a direct bearing on the amount of computer time required for each component solution.

II: For N ODD

For this case, the equivalent equations to A1.2 and A1.4 are

$$\begin{aligned} h(n) &= a(n) + b(n), \quad n=0,\ldots,(N-3)/2 \\ h(\tfrac{N-1}{2}) &= a(\tfrac{N-1}{2}) + b(\tfrac{N-1}{2}) \\ h(N-1-n) &= a(n) - b(n), \quad n=0,\ldots,(N-3)/2 \end{aligned} \qquad \text{A1.12}$$

where the term $h(\frac{N-1}{2})$ maps to itself and in practice $b(\frac{N-1}{2})=0$.

Hence,

$$H(\omega) = e^{-j\omega T(N-1)/2} \left\{ a(\tfrac{N-1}{2}) + \sum_{n=0}^{(N-3)/2} 2a(n)\cos\left[\omega T[n-(N-1)/2]\right] - j \sum_{n=0}^{(N-3)/2} 2b(n)\sin\left[\omega T[n-(N-1)/2]\right] \right\} \qquad \text{A1.13}$$

In this case, there is an extra term $a(\frac{N-1}{2})$ in A1.13 compared with A1.4. To make A1.13 collapse to two terms so that the same algorithm can be used, define

$$\begin{aligned} 2a(n) &= c(n), \quad n=0,\ldots,(N-3)/2 \\ a(\tfrac{N-1}{2}) &= c(\tfrac{N-1}{2}) \end{aligned} \qquad \text{A1.14}$$

then A1.13 becomes

$$H(\omega) = e^{-j\omega T(N-1)/2} \left\{ \sum_{n=0}^{(N-1)/2} c(n)\cos\left[\omega T[n-(N-1)/2]\right] - j \sum_{n=0}^{(N-3)/2} 2b(n)\sin\left[\omega T[n-(N-1)/2]\right] \right\} \qquad \text{A1.15}$$

Now the identical procedure used in the N even case can be followed and equivalent equations to A1.7, A1.9, A1.10 developed. However, in this case, each component solution is returned in c(n) and b(n). The extra step required to produce h(n) is

$$h(n) = \frac{c(n)}{2} + b(n) , \qquad n=0,\ldots,(N-3)/2$$

$$h(\frac{N-1}{2}) = c(\frac{N-1}{2}) \qquad \text{A1.16}$$

$$h(N-1-n) = \frac{c(n)}{2} - b(n) , \qquad n=0,\ldots,(N-3)/2.$$

In performing the optimisation on the components of A1.7, recall that the frequency response $H_b(\omega)$ of an odd symmetric sequence b(n) must be zero at $\omega=0$. Thus, the order of the optimisation procedure for A1.7b, A1.9 and A1.10 can be reduced to $k = 1,\ldots,M$. However, for some reason that was never properly established, the Harwell routine VA07A would not behave when this reduction was attempted.

SUBROUTINE CUTHBERT

For the sake of completeness, the source code developed to implement the procedure described above is presented in Chapter 6. The actual optimisation code routine VA07A is an Harwell Library Subroutine and is specifically designed to solve the problem of A1.6 [Fletcher, 1971].

In section 3.2 when describing Cuthbert's approach, it was pointed out that he found that by performing a second optimisation run on each component over the stopband specification only, some further improvement was obtained. The code below provides an option to implement this feature if desired.

APPENDIX 2

DERIVATION OF THE EQUATION DESCRIBING THE IMPULSE RESPONSE OF A FIR FILTER WITH A SINE PHASE DEVIATION.

Given the causal lowpass filter frequency specification (chapter 3, Fig.3.1)

$$A(\omega) = \begin{cases} 1.0 & , \ 0 \leq \omega \leq \omega_c \\ 0.0 & \text{otherwise} \end{cases} \qquad \text{A2.1a}$$

$$\theta_T(\omega) = \begin{cases} \omega t_o - b\sin\left(\frac{2\pi\omega}{\omega_o}\right) & , \ 0 \leq \omega \leq \omega_o \\ \omega t_o & \omega_o \leq \omega \leq \omega_c \end{cases} \qquad \text{A2.1b}$$

and $H(\omega) = A(\omega)e^{-j\theta(\omega)}$ then,

$$h(t) = \frac{1}{\pi}\int_0^{\pi} H(\omega)e^{j\omega t}\, d\omega$$

Since h(t) is real, the quadrature terms in jsin in A2.2 vanish, and

$$h(t) = \frac{1}{\omega_c-\omega_o}\int_{\omega_o}^{\omega_c} e^{-j\omega t_o}e^{j\omega t}\, d\omega + \frac{1}{\omega_o}\int_0^{\omega_o} e^{-j\left(\omega t_o - b\sin\frac{2\pi\omega}{\omega_o}\right)}\, e^{j\omega t}\, d\omega \qquad \text{(A2.3)}$$

Using $\tau = (t-t_o)$, the first integral in (A2.3) has solution

$$h_1(t) = \frac{1}{(\omega_c-\omega_o)\tau}\left(\sin(\tau\omega_c) - \sin(\tau\omega_o)\right) \qquad \text{A2.4}$$

Considering the second integral, re-arrange to

$$h_2(t) = \frac{1}{\omega_o}\int_0^{\omega_o} e^{j\omega\tau}\, \Delta\theta(\omega)\, d\omega \qquad \text{A2.5}$$

where $\Delta\theta(\omega) = e^{jb\sin\left(\frac{2\pi\omega}{\omega_o}\right)}$

$\Delta\theta(\omega)$ is periodic in ω_o. Thus $\Delta\theta(\omega)$ can be expanded in a Fourier series in the interval $(-\omega_o/2, \omega_o/2)$. Hence

$$\Delta\theta(\omega) = \sum_{n=-\infty}^{\infty} \alpha_n\, e^{j\frac{n\pi\omega}{\omega_o/2}} \qquad \text{A2.6}$$

where $$\alpha_n = \frac{1}{\omega_o} \int_{-\omega_o/2}^{\omega_o/2} \Delta\theta(\omega) e^{-j2n\pi\omega/\omega_o} \, d\omega$$

$$= \frac{1}{\omega_o} \int_{-\omega_o/2}^{\omega_o/2} e^{j\left(b\sin(2\pi\omega/\omega_o) - j2n\pi\omega/\omega_o\right)} \, d\omega \qquad \text{A2.7}$$

Substituting $x = 2\pi\omega/\omega_o$, then

$$\alpha_n = \frac{1}{2\pi} \int_{-\pi}^{\pi} e^{j(b\sin(x) - nx)} \, dx \qquad \text{A2.8}$$

It is well known that the integral in A2.8 equals the Bessel function $J_n(b)$ of the first kind. Thus $\alpha_n = J_n(b)$. Hence, substituting for α_n in A2.6 and then A2.6 in A2.5

$$h_2(t) = \frac{1}{\omega_o} \int_0^{\omega_o} e^{j\omega\tau} \sum_{n=-\infty}^{\infty} J_n(b) \, e^{j(2n\pi\omega/\omega_o)} \, d\omega$$

$$= \frac{1}{\omega_o} \sum_{n=-\infty}^{\infty} J_n(b) \int_0^{\omega_o} e^{j\left(\tau + \frac{2n\pi}{\omega_o}\right)\omega} \, d\omega$$

$$= \frac{1}{\omega_o} \sum_{n=-\infty}^{\infty} J_n(b) \frac{\sin\left[\tau+(2\pi n/\omega_o)\right]\omega_o}{\left(\tau+(2\pi n/\omega_o)\right)\omega_o} \qquad \text{A2.9}$$

whence, using A2.4 and A2.9:

$$h(t) = h_1(t) + h_2(t)$$

APPENDIX 3

DERIVATION OF THE EQUATION DESCRIBING THE IMPULSE RESPONSE OF A FIR FILTER WITH A SQUARE WAVE PHASE DEVIATION.

Given the causal lowpass filter frequency specification (chapter 3, Fig. 3.3)

$$A(\omega) = \begin{cases} 1.0 \ , \ 0 \leq \omega \leq \omega_c \\ 0.0 \quad \text{otherwise} \end{cases} \qquad \text{A3.1a}$$

$$\theta_T(\omega) = \begin{cases} \omega t_o + b \ , & 0 \leq \omega \leq \omega_c/2 \\ \omega t_o - b \ , & \omega_c/2 < \omega \leq \omega_c \\ \omega t_o \ , & \end{cases} \qquad \text{A3.1b}$$

and $H(\omega) = A(w)e^{-j\theta_T(\omega)}$, then

$$h(t) = \frac{1}{\pi}\int_0^{\pi} H(\omega)e^{j\omega t}\, d\omega$$

$$= \frac{2}{\omega_c}\int_0^{\omega_c/2} e^{-j[\omega t_o+b]}e^{j\omega t}\, d\omega + \frac{1}{\omega_c}\int_{\omega_c/2}^{\omega_c} e^{-j[\omega t_o-b]}\, e^{j\omega t}\, d\omega \qquad \text{(A3.2)}$$

Since h(t) is real, the quadrature terms in jsin in A3.2 vanish. Expanding the cosine terms with $\tau=(t-t_o)$ and integrating

$$h(t) = \frac{2}{\omega_c\tau}\cos b\sin(\omega_c\tau/2) + \sin b\left(\cos(\omega_c\tau/2) - 1\right)$$

$$+ \frac{2}{\omega_c\tau}\cos b\left(\sin(\omega_c\tau) - \sin(\omega_c\tau/2)\right)$$

$$- \frac{2}{\omega_c\tau}\sin b\left(\cos\omega_c\tau) - \cos(\omega_c\tau/2)\right) \qquad \text{A3.3}$$

Collecting terms,

$$h(t) = \frac{2}{\omega_c\tau}\cos b\, \sin(\omega_c\tau) + \sin b\left(2\cos(\omega_c\tau/2) - \cos(\omega_c\tau) - 1\right) \qquad \text{A3.4}$$

Using the trigonometric identity $\cos 2\theta = 1-2\sin^2\theta$, after some arithmetic,

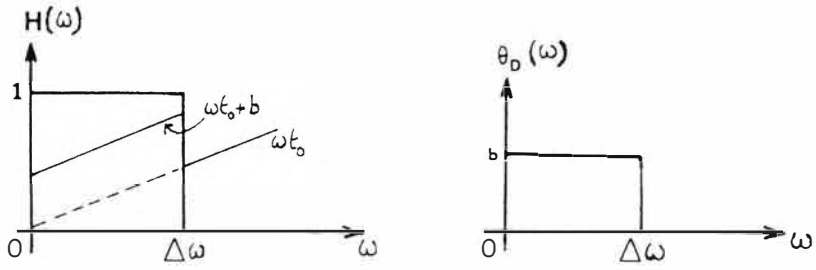
$$h(t) = 2\cos b\frac{\sin(\omega_c\tau)}{\omega_c\tau} + \sin b\left(\frac{\sin^2(\omega_c\tau/2)}{\omega_c\tau/2} - \frac{\sin^2(\omega_c\tau/4)}{\omega_c\tau/4}\right) \qquad \text{A3.5}$$

where the presentation of equation A3.5 has been deliberately chosen to emphasise the sinx/x functions.

APPENDIX 4

DERIVATION OF THE EQUATION DESCRIBING THE IMPULSE RESPONSE OF A PROTOTYPE LOWPASS FILTER WITH A CONSTANT PHASE DEVIATION.

Given the causal lowpass filter frequency specification:



$$A(\omega) = \begin{cases} 1.0\ , & 0 \leq \omega \leq \Delta\omega \\ 0.0 & \text{otherwise} \end{cases} \qquad \text{A4.1a}$$

$$\theta_T(\omega) = \begin{cases} \omega t_o + b, & 0 \leq \omega \leq \Delta\omega \\ \omega t_o & \text{otherwise} \end{cases} \qquad \text{A4.1b}$$

and $H(\omega) = A(\omega)e^{-j\theta_T(\omega)}$

Then,

$$h_p(t) = \frac{1}{\pi}\int_0^{\pi} H(\omega)e^{j\omega t}\ d\omega \qquad \text{A4.2}$$

Since $h_p(t)$ is real, the quadrature terms in A4.2 vanish and using $\tau=(t-t_o)$

$$h_p(t) = \frac{1}{\Delta\omega}\int_o^{\Delta\omega} \cos(\omega\tau - b)\ d\omega$$

$$= \frac{1}{\Delta\omega\tau}(\sin(\Delta\omega\tau - b) - \sin b)$$

$$= \frac{1}{\Delta\omega\tau}(\cos b\sin(\Delta\omega\tau) - \sin b[1-\cos(\Delta\omega\tau)]) \qquad \text{A4.3}$$

Using the relation $\cos 2\phi = 1-2\sin^2\phi$ in A4.3, after some arithmetic,

$$h_p(t) = \cos\left(\frac{\Delta\omega\tau}{2} + b\right)\frac{\sin(\Delta\omega\tau/2)}{\Delta\omega\tau/2} \qquad \text{A4.4}$$

The arrangement of the result in A4.4 has been used deliberately to emphasise the sinx/x function.

APPENDIX 5.0

3-SINUSOID EXAMPLE: ESTABLISHING A SET OF HARMONIC PHASES OF A SIGNAL THAT HAS ITS ANALYTIC ZEROS EVENLY DISTRIBUTED IN TIME.

In section 5.4.1, a solution to the quadratic equation

$$c_1 + c_2 x + c_3 x^2 = 0 \qquad \text{(A5.1)}$$

with $c_1 = 0.6 \angle -\pi/2$

$c_2 = 1.0 \angle -\pi/2$

$c_3 = 0.8 \angle -\pi/2$

where $\angle c = \text{Arg}(c)$

is required. Subsequent application of equation (4.16) gives the analytic zeros

$$\sigma_k = \frac{1}{\Omega} \ln|x_k| \qquad \text{(A5.2a)}$$

$$\tau_k = \frac{1}{\Omega} \text{Arg}(x_k) \qquad \text{(A5.2b)}$$

It is suggested that the τ_k be evenly spaced in the period $T=2\pi/\Omega$. Ignoring the scaling $1/\Omega$ terms, such an equidistant spacing for this quadratic (two zeros) requires

$$\tau_2 - \tau_1 = \pi \qquad \text{(A5.3a)}$$

$$\text{or} \quad \tan(\tau_2) - \tan(\tau_1) = 0 \qquad \text{(A5.3b)}$$

Representing the solution to equation (A5.1) as

$$x_k = x_{rk} + jx_{qk} \qquad \text{(A5.4)}$$

and then using equation (A5.4) and (A5.2b) in (A5.3b) requires

$$\frac{x_{q2}}{x_{r2}} - \frac{x_{q1}}{x_{r1}} = 0 \qquad \text{(A5.5)}$$

The solution to equation (A5.1) is

$$x = -\frac{c_2}{2c_3} \pm \frac{1}{2c_3}\sqrt{(c_2^2 - 4c_1c_3)}$$

$$= -\frac{c_2}{2c_3} \pm \frac{j}{2c_3}\sqrt{(4c_1c_3 - c_2^2)} \qquad \text{(A5.6)}$$

From equation (A5.6), it is apparent that $x_{r1} = x_{r2}$. Thus, equation

(A5.5) collapses to

$$x_{q2} - x_{q1} = 0 \qquad \text{(A5.7)}$$

Similarly from equation (A5.6), $x_{q2} = -x_{q1}$, so the only way to satisy equation (A5.7) is for $x_{qk} = 0$, k=1,2.

Thus, for this general quadratic example, equidistant spacing of the analytic zeros requires that the real signal (from which the polynomial of equation (A5.1) is derived) possess only real zeros - ie zero crossings. Hence

$$4c_1c_3 - c_2^2 = 0 \qquad \text{(A5.8)}$$

and expanding this equation in terms of magnitude and angle gives

$$4|c_1||c_3| \angle c_1 + c_3 = |c_2|^2 \angle 2c_2 \qquad \text{(A5.9)}$$

Clearly, the values of c_k in equation (A5.1) do not satisfy equation (A5.9). Furthermore, given that modification of $|c_k|$ is prohibited, there are a denumerable number of combinations of $\text{Arg}(c_k)$ that will satisfy equation (A5.9). Writing

$$\begin{aligned} c_1 &= a+jb = 0-j0.6 \\ c_2 &= c+jd = 0-j1.0 \\ c_3 &= e+jf = 0-j0.8 \end{aligned} \qquad \text{(A5.10)}$$

equation (A5.8) becomes

$$4[ae-bf+j(be+af)] = c^2-d^2 + j2cd \qquad \text{(A5.11)}$$

Choosing one particular solution at random: retain c_2, c_3 and modify c_1. Equating real and imaginary parts of equation (A5.11) gives

$$4(ae-bf) = c^2-d^2$$

or $-4b'(0.8) = -(1.0)$

so $b' = 0.3125$ (A5.12a)

and $be-af = 2cd = 0$ because $a=c=e=0$ (A5.12b)

Modification of $|c_1|$ is prohibited, so

$$a' = \sqrt{(|c_1|-b'^2)} = 0.5122 \qquad \text{(A5.13)}$$

which gives

$$\text{Arg}(c_1) = \tan^{-1}\left(\frac{0.3125}{0.5122}\right) = 0.8642 \text{ radians.} \qquad \text{(A5.14)}$$

The final step is to compute θ_1 required by equation (5.9) of section 5.4.1. In this section, it was indicated that equation (5.9) appeared

as a negative sine series in the FFT which results in $-\pi/2$ as the result for $\text{Arg}(c_k)$. Hence, adding back this $\pi/2$ to the result of equation (A5.14) gives

$$\theta_1 = 0.862+\pi/2 = 2.1186 \text{ radians.}$$

APPENDIX 6

VAX FORTRAN subroutines that implement some of the design algorithms discussed in chapters 2 and 3. The subroutines included here are

SUBROUTINE DOLP_CHEB

SUBROUTINE KAISER

SUBROUTINE CUTHBERT

SUBROUTINE ALGAZI

SUBROUTINE EVEN

The subroutine EVEN is referenced by the four preceding modules.

```
C-----------------------------------------------------------------------
C                SUBROUTINE DOLPHCHEB
C
C        AUTHOR:    R.D.C. Stephen
C                   Dept. of Electrical and Electronic Engineering
C                   University of Canterbury, Private Bag, New Zealand.
C
C        LANGUAGE: VAX FORTRAN, Version 4.3
C                   note: Vax fortran is a superset of FORTRAN-77
C
C        USAGE:     CALL DOLPHCHEB(X,Y,NFFT,ALPHA,HIMP,TEMP)
C
C        PARAMETERS:
C
C            X,Y    REAL*4 arrays of dimension NFFT used with the FFT
C                   routine.
C
C            NFFT   INTEGER*4. Length of arrays and used by FFT routine.
C
C            ALPHA REAL*4 parameter for calculating the frequency
C                         response of the Dolph-Chebyshev window.
C
C            HIMP   INTEGER*4 is the required number of impulse
C                             response coefficients of the filter.
C
C            TEMP   REAL*4 array of dimension NFFT used for temporary
C                         storage.
C
C  DATA FORMAT:
C
C    The following comments are based on the use of the FFT842 routine:
C
C    The routine is based on the data arrangement that ignores the
C    linear phase term. That is, the routine expects the +ve
C    frequencies in the lower 1 - NFFT/2+1 points and the -ve
C    frequencies in the upper NFFT/2+2 - NFFT points in the array X.
C    On exit, the lower half of the impulse response resides in the
C    NFFT/2+2 - NFFT points and the upper half of the impulse response
C    in the 1 - NFFT/2+1 points.
C
C  DESCRIPTION:
C
C    On entry to the routine, the "desired" frequency response (the
C  model) is held in the X,Y arrays. The routine does an inverse FFT,
C  truncates the impulse response to HIMP and stores it temporarily in
C  TEMP. Then the frequency response of the Dolph-Chebyshev window
C  corresponding to a parameter value ALPHA and (time) sequence length
C  HIMP is computed, loaded in X and another inverse FFT performed to
C  generate the window coefficients. The impulse response coefficients
C  in TEMP are then multiplied into the window coefficients in X, Y is
C  set zero and the routine returns with the result in X.
C
C  Note: in situations requiring only linear phase filters, this
C        routine contains redundant information. In practice the FFT
C        routine FTT842 should be replaced by one using only real data
C        with the +ve half of the frequency domain information. In
C        this case, the array TEMP is not required, X, Y and NFFT can
C        be reduced to NFFT/2 in length, with Y used as the temporary
C        storge. Use of the symmetry property on the result completes
C        the required impulse response.
```

```
C
C  SUBROUTINES REFERENCED:
C
C         FFT842 - from IEEE programs for Digital Signal Processing.
C
C-----------------------------------------------------------------------
      SUBROUTINE DOLPHCHEB(X,Y,NFFT,ALPHA,HIMP,TEMP)

      PARAMETER PI=3.1415926
      INTEGER   NFFT, HIMP
      INTEGER   I, J, K
      REAL      X(NFFT), Y(NFFT), TEMP(NFFT), ALPHA
      REAL      BETA, DEN, FRAC, Z
C
C  get ideal impulse response, truncate it to HIMP and hold in TEMP
C  and clear Y array
C
      CALL FFT842(1,NFFT,X,Y)
      K=HIMP/2
      DO I=K+2,NFFT-K
        X(I)=0.0
      END DO
      DO I=1,NFFT
        TEMP(I)=X(I)
        Y(I)=0.0
      END DO
C
C  compute +ve half of frequency response of Dolph-Chebyshev window
C  and load in X. Mirror image for -ve half. Then do IFFT.
C
      DEN=10.0**ALPHA
      BETA=COSH(ALOG(DEN+SQRT(DEN**2-1.0))/FLOAT(HIMP))
      FRAC=PI/FLOAT(NFFT)
      K=1
      DO I=0,NFFT/2
        Z=BETA*COS(FRAC*FLOAT(I))
        IF(ABS(Z).GE.1.0) THEN
          X(I+1)=COSH(FLOAT(HIMP)*ALOG(Z+SQRT(Z**2-1.0)))/DEN
        ELSE
          Z1=Z/SQRT(1.0-Z**2)
          X(I+1)=COS(FLOAT(HIMP)*(PI/2.0-ATAN(Z1)))/DEN
        END IF
        K=K+1
      END DO
      J=2
      DO I=NFFT,NFFT/2+2,-1                !mirror image for -ve f's
        X(I)=X(J)
        J=J+1
      END DO
C
C  parseval: normalise window power so get unity power gain for final
C  sequence
C
      Z=0.0
      DO I=1,NFFT
        Z=Z+X(I)
      END DO
      Z=Z/FLOAT(NFFT)
```

```
      DO I=1,NFFT
        X(I)=X(I)/Z
      EN DDO
      CALL FFT842(1,NFFT,X,Y)
C
C  now multiply window coefficients with truncated impulse response
C  in TEMP and clear Y. Final result lies in X.
C
      DO I=1,NFFT
        X(I)=X(I)*TEMP(I)
        Y(I)=0.0
      END DO
      RETURN
      END
```

```
C-----------------------------------------------------------------------
C              SUBROUTINE KAISER
C
C       AUTHOR:   R.D.C. Stephen
C                 Dept. of Electrical and Electronic Engineering,
C                 University of Canterbury, Christchurch
C                 New Zealand.
C
C       LANGUAGE: VAX FORTRAN, Version 4.3
C
C       USAGE:    CALL KAISER(X,Y,NFFT,ATTEN,HIMP)
C
C       PARAMETERS:
C
C           X,Y   REAL*4 arrays of dimension NFFT used with the FFT
C                 routine.
C
C           NFFT  INTERGER*4 dimension of FFT arrays
C
C           ATTEN REAL*4 required magnitude of the stopband
C                 attenuation in dB. (as a POSITIVE number)
C
C           HIMP  INTEGER*4 the required length of the impulse response.
C                 HIMP < NFFT/2
C
C  DATA FORMAT:
C
C    The following comments are based on the use of the FFT842 routine:
C
C    The routine is based on the data arrangement that ignores the
C    linear phase term. That is, the routine expects the +ve
C    frequencies in the lower 1 - NFFT/2+1 points and the -ve
C    frequencies in the upper NFFT/2+2 - NFFT points in the array X.
C    On exit, the lower half of the impulse response resides in the
C    NFFT/2+2 - NFFT points and the upper half of the impulse response
C    in the 1 - NFFT/2+1 points.
C
C  DESCRIPTION:
C
C    On entry to the routine, the "desired" frequency response (the
C  model) is held in the X,Y arrays. The routine does an inverse FFT
C  and then truncates the result to HIMP. The window coefficients are
C  computed and multiplied into the truncated impulse response.
C  Finally Y is set to zero and the routine returns with the h(n) in
C  X.  The ATTEN parameter is that required by equation 7 of Kaiser's
C  paper "The Io-sinh windows
C
C  Note: in situations requiring only linear phase filters, this
C        routine contains redundant information. In practice the FFT
C        routine FFT842 should be replaced by one using only real data
C        (X) with the +ve half of the frequency domain information
C        only. In this case, array Y is not required and X and NFFT
C        can be reduced to NFFT/2 in length. Use of the symmetry
C        property on the result completes the required impulse
C        response.
```

```
C
C  SUBROUTINES REFERENCED:
C
C       FFT842 - from IEEE Programs for Digital Signal Processing
C       EVEN
C       FUNCTION BIO
C-----------------------------------------------------------------------
        SUBROUTINE KAISER(X,Y,NFFT,ATTEN,HIMP)

        PARAMETER PI=3.1415926
        INTEGER   NFFT, HIMP
        INTEGER   I, J, K, L
        REAL      X(NFFT), Y(NFFT), ALPHA, ATTEN
        REAL      BIO, DEN, TOP, Z
        LOGICAL*1 EV
C
C  if HIMP is even, first offest F-domain data by T/2
C
        EV=.FALSE.
        ATTEN=ABS(ATTEN)
        IF(MOD(HIMP,2).EQ.0) EV=.TRUE.
        IF(EV) CALL EVEN
C
C  get ideal impulse response, truncate it to HIMP
C
        CALL FFT842(1,NFFT,X,Y)
        J=HIMP/2+1
        K=NFFT+1-J
        IF(.NOT.EV)J=J+1
        DO I=J,K
          X(I)=0.0
        END DO
C
C  setup parameters for window and do it. Use equation (7) from
C  his paper.
C
        IF(ATTEN.GT.50.0) THEN
          ALPHA=(ATTEN-8.7)*0.1102
        ELSE IF(ATTEN.GT.21.0) THEN
          Z=ATTEN-21.0
          ALPHA=0.5842*Z**0.4+0.07886*Z
        ELSE
          ALPHA=0.0
        END IF
        J=HIMP/2
        IF(.NOT.EV) J=J+1
        DEN=BIO(ALPHA)
        DO I=1,J
          AINDEX=FLOAT(I)
          K=I+1
          L=NFFT+1-I
          IF(EV) THEN
            K=K-1
            AINDEX=AINDEX-0.5
          END IF
          TOP=ALPHA*SQRT(1.0-(AINDEX/FLOAT(J))**2)
          X(K)=X(K)*BIO(TOP)/DEN
          X(L)=X(L)*BIO(TOP)/DEN
        END DO
```

```
      DO I=1,NFFT
        Y(I)=0.0
      END DO
      RETURN
      END
C-----------------------------------------------------------------------
C       REAL FUNCTION BIO
C
C       This function routine is the subroutine due to Kaiser shown
C   in the text Theory and Application
C   of Digital Signal Processing by Rabiner and Gold, section 3.14,
C   p 103.
C       This function computes the besel function of order zero for
C   values of X from 0.0 to 20.0 inclusive
C
C       Called By: KAISER
C
C       Calls:          nothing
C
C        Input:       X - the input value
C-----------------------------------------------------------------------
      REAL FUNCTION BIO(X)
C  X must be between 0.0 and 20.0
      Y=X/2.0
      T=1.0E-08
      E=1.0
      DE=1.0
      DO 1 I=1,25
        DE=DE*Y/FLOAT(I)
        SDE=DE**2
        E=E+SDE
       IF(E*T-SDE) 1,1,2
 1    CONTINUE
 2    BIO=E
      END
```

```
C-----------------------------------------------------------------------
C            SUBROUTINE CUTHBERT
C
C  AUTHOR: R.D.C. STEPHEN
C          DEPT. OF ELECTRICAL AND ELECTRONIC ENGINEERING
C          UNIVERSITY OF CANTERBURY
C          PRIVATE BAG
C          CHRISTCHURCH, NEW ZEALAND.
C
C  DATE:   SEPTEMBER 1986
C
C  LANGUAGE:  VAX FORTRAN, VERSION 4.3
C           (note: VAX Fortran is a superset of FORTRAN-77)
C
C  DESCRIPTION:
C      Uses Cuthbert's method [Cuthbert,1974a; Cuthbert,1974b] to
C  compute the impulse response of linear or non-linear phase,
C  ordinary or Hilbert, lowpass and bandpass filters. The method
C  uses a least-mean-square approach with a mathematical optimisation
C  routine [Fletcher, 1971].
C      On entry to the routine, the desired frequency specification
C  is expected in X and Y. These are the FFT arrays used in the outer
C  program as the commmon base. Both the real X and quadrature Y
C  components are copied to TEMPR and TEMPQ as temporary storage and
C  for use in RESID.
C      Then Y is zeroed and an inverse FFT computed to get the
C  starting approximation for the real part of the impulse response.
C      After the real part has been optimised (the a(n)), this is held
C  temporarily in TEMPR overwriting the original data. Then X is
C  zeroed and the quadrature part of the frequency specification is
C  restored to Y. An inverse FFT then provides the initial
C  approximation for the quadrature optimisation part.
C      On completion, the routine exits with the complete impulse
C  response in X with Y=0
C      The FFT routine used is FFT842 from the IEEE Programs for
C  Digital Signal Processing package. Thus NFFT must be a power of 2.
C
C  USAGE:  CALL CUTHBERT(X,Y,TEMPR,TEMPQ,NFFT,OPTION,JTYPE,FTYPE,
C                        WEIGHTP,WEIGHTS,HIMP,ITER)
C
C  PARAMETERS:
C
C     X,Y     : REAL*4, arrays of dimension NFFT. These are the FFT
C               arrays of the calling program. On entry, they contain
C               the desired frequency response. On exit X contains the
C               required impulse response and Y=0
C
C     TEMPR   : REAL*4, arrays of dimension NFFT used for temporary
C     TEMPQ   : storage
C
C     NFFT    : INT*4, dimension of X,Y, TEMPQ. Maximum value of 1024
C               allowed (see notes below)
C
C     OPTION  : INT*4,
C               OPTION=0 gives a single pass through the optimisation
C                        procedure.
C               OPTION=1 gives a second pass through the optimisation
C                        procedure over the stopband specification only.
```

```
C
C     JTYPE  : INT*4,
C              =0 ordinary linear phase filter
C              =1 Hilbert linear phase filter
C              =2 non-linear phase filter
C
C     FTYPE  : INT*4,
C              =0 lowpass  filter
C              =1 bandpass filter
C
C
C     WEIGHTP: REAL*4, is the weighting value in the defined passband.
C
C     WEIGHTS: REAL*4, is the weighting value in the defined stopband.
C
C       HIMP : INT*4, the required length of the impulse response
C
C       ITER : INT*4, the number of iterations of the optimisation
C                     algorithm to perform
C
C  NOTES:
C
C  Section 7 of the VA07A calling document allows the routine to be
C  adjusted to handle different order problems (MxN). The code here
C  uses a parameter statement with MMAX and NMAX to setup the maximum
C  problem size that this code can handle. It is presently set to
C  MMAX=512, NMAX=300. MMAX corresponds to half the FFT array lengths
C  that are passed (NFFT). Thus, the maximum FFT array length that can
C  be passed to this routine is 2*MMAX without re-dimensioning MMAX.
C  The single parameter statement dimensions all the required working
C  arrays at the start.
C
C  If MMAX, NMAX are altered, the array declarations for DR and WEIGHT
C  in modules RESID and LSQ must be altered to match the new values.
C
C  In order to facilitate an integrated, single call, the argument calling
C  lit of VA07A, RESID and LSQ have been augmented beyond the original
C  documentation to include TEMPR, TEMPQ and NFFT
C
C  SUBROUTINES REFERENCED:
C
C       VA07A  - Harwell Library routine: optimisation code
C       RESID  - user written routine required by VA07A
C       LSQ    - user written routine required by VA07A
C       MA10A  - Harwell routine; called by VA07A
C       FM02AS - REAL*8 function; Harwell routine; called by LSQ
C       FFT842 - complex FFT routine from IEEE Programs for Digital
C                Signal Processing
C
C    WARNING: modules VA07A, MA10A, FM02AS are FORTRAN-IV standard
C             code. These modules MUST be compiled using the
C             /NOF77 qualifier otherwise the program will crash when
C             FM02AS is invoked. Other, unexplained "funnies" also occur
C             in VA07A if this qualifier is not used.
C
C             FFT842 is also FORTRAN-IV standard, but to date has
C             performed ok with compilation using the default /F77
```

```
C
C             modules CUTHBERT, RESID, LSQ are VAX FORTRAN standard
C             and should be compiled using /F77/OPTIMISE qualifiers
C-----------------------------------------------------------------------
      SUBROUTINE CUTHBERT(X,Y,TEMPR,TEMPQ,NFFT,OPTION,JTYPE,FTYPE,
     +                    WEIGHTP,WEIGHTS,HIMP,ITER)
C
      PARAMETER MMAX=512, NMAX=300, PI=3.1415926
C
C  local variables.
C
      INTEGER   NFFT, JTYPE, FTYPE, HIMP, ITER
      REAL      X(NFFT), Y(NFFT), TEMPR(NFFT), TEMPQ(NFFT), WEIGHTP,
     +          WEIGHTS, WEIGHT(MMAX), Z
      INTEGER   I, M, N
      LOGICAL*1 ODD
C
C  variables associated with VA07A (note X has been changed to VEC to
C  avoid conflict with local array X):
C
      REAL     R(MMAX), A(NMAX,NMAX), SS, TOL
      INTEGER  MS, IPRINT, MAXFN, MODE, LOOP, NF1, NF2, NF3, NF4
      INTEGER  H2IMP, F2TYPE
C
C  RESID and LSQ are the subroutines referenced in VA07A argument list.
C  /USERA/ shared with module LSQ
C  /USERB/ shared with module RESID
C  /OPT/ and /VA07C/ are shared with module TOEPLITZ to save space
C
      COMMON/VA07B/S(NMAX)
      COMMON/VA07C/T(NMAX)
      COMMON/VA07D/U(NMAX)
      COMMON/VA07E/V(NMAX)
      COMMON/VA07F/W(MMAX)
      COMMON/OPT/ VEC(NMAX), D(NMAX), EPS(NMAX)
      COMMON/USERA/ DR(MMAX,NMAX)
      COMMON/USERB/ FINCR, LOOP, WEIGHT, ODD, H2IMP, F2TYPE,
     +              NF1, NF2, NF3, NF4
      EXTERNAL RESID, LSQ
C
      H2IMP=HIMP
      F2TYPE=FTYPE
C
C first check that NFFT and HIMP are not too big
C
      IF(NFFT/2.GT.MMAX) THEN
        WRITE(6,'(X,A56)') ' ERROR: NUMBER OF INTERPOLATING FREQUENCIES
     + IS TOO LARGE'
        WRITE(6,'(X,A41)') ' (NFFT MUST BE LESS THAN OR EQUAL TO 1024)'
        RETURN
      END IF
      IF(HIMP.GT.NMAX) THEN
        WRITE(6,'(X,A60)') ' ERROR: NUMBER OF IMPULSE RESPONSE COEFFICIENTS
     + IS TOO LARGE'
        WRITE(6,'(X,A41)') ' (HIMP MUST BE LESS THAN OR EQUAL TO 300)'
        RETURN
      END IF
```

```
      ODD=.FALSE.
      IF(MOD(HIMP,2).NE.0) ODD=.TRUE.
C
C  for module VA07A: set M, N, MODE, IPRINT, MAXFN
C
C     IPRINT=-1    !show results every iteration and exclude R()
      IPRINT=0     !don't show results
      N=HIMP/2
      IF(ODD) N=N+1
      M=NFFT/2
      FINCR=PI/FLOAT(M-1)
      MAXFN=ITER+1
C     MODE=1
      MODE=2       !algorithm is better behaved with this option
C
C  initialise A(i,j), and copy frequency domain spec to TEMPR and TEMPQ
C
      DO J=1,N
        DO I=1,N
          A(I,J)=0.0
        END DO
      END DO
      DO I=1,NFFT
        TEMPR(I)=X(I)
        TEMPQ(I)=Y(I)
      END DO
C
C  now set required tolerance on accuracy of solution VEC arbitrarily to be
C  1.0E-6
C
      DO I=1,N
        EPS(I)=1.0E-6
      END DO
C
C load WEIGHT array template and hence setup passband delimiter
C indices NF1, NF2. These are used in module RESID
C
      DO I=1,M
        Z=SQRT(X(I)**2+Y(I)**2)
        IF(Z.GT.0.0) THEN
          WEIGHT(I)=WEIGHTP
        ELSE
          WEIGHT(I)=WEIGHTS
        END IF
      END DO
      J=1
      K=1
      Z=WEIGHTP
      IF(FTYPE.EQ.1) THEN
        K=K+1
        Z=WEIGHTS
      END IF
      DO I=1,K
        DO WHILE(WEIGHT(J).EQ.Z)  !stopband of bandpass or passband of lowpass
          J=J+1
        END DO
```

```
        IF(I.EQ.1) NF1=J-1
        IF(I.EQ.2) NF3=J-1
        IF(Z.EQ.WEIGHTP) THEN  !next frequency interval
          Z=WEIGHTS
        ELSE
          Z=WEIGHTP
        END IF
        IF(I.EQ.1) NF2=J
        IF(I.EQ.2) NF4=J
      END DO
C
C  now setup indices needed for LOOP= 1,2 for ordinary linear or non-
C  linear phase, or for LOOP =3,4 for Hilbert linear or non-linear
C  phase filters. M is the number of frequency harmonics (ie the
C  order of the approximation problem) passed to VA07A for the first
C  pass (LOOP = 1 or 3). MS is the order of the approximation problem
C  for the stopband only pass if requested (LOOP = 2 or 4)
C
        IF(OPTION.EQ.1) THEN
          IF(FTYPE.EQ.0) THEN
            MS=NFFT/2-NF2+1
          ELSE
            MS=NF1+(NFFT/2-NF2)+1
          END IF
        END IF
C
C  BEGIN COMPUTATIONS:
C  Real Part: Get initial approximation of a(n) by making Y=0, do IFFT
C  and then copy first N values from X to VEC and optimise. Copy result
C  into TEMPR as temporary storage. LOOP is used in the modules RESID
C  and LSQ associated to control the data selected.
C
      IF(JTYPE.EQ.0.OR.JTYPE.EQ.2) THEN
        WRITE(6,'(X,A11)') ' Real Part:'
        DO I=1,NFFT
          Y(I)=0.0
        END DO
        CALL FFT842(1,NFFT,X,Y)
        IF(ODD) THEN                    !starting estimate
          J=N
          DO I=1,N
            VEC(J)=X(I)
            J=J-1
          END DO
        ELSE
          DO I=1,N
            VEC(I)=X(I)
          END DO
        END IF
        WRITE(6,'(X,A11)') ' First pass'
        LOOP=1
        CALL VA07A(RESID,LSQ,M,N,VEC,R,SS,A,D,EPS,IPRINT,MAXFN,MODE,
     +             TEMPR,TEMPQ,NFFT)
C
C  stopband pass
C
```

```
        IF(OPTION.EQ.1) THEN
          WRITE(6,'(X,A19)') ' Stopband only pass'
          LOOP=2
          CALL VA07A(RESID,LSQ,MS,N,VEC,R,SS,A,D,EPS,IPRINT,MAXFN,MODE,
     +               TEMPR,TEMPQ,NFFT)
        END IF
C
C  result in VEC comes back in reverse order - ie h(0) = VEC(N)
C
        DO I=1,N
          TEMPR(I)=VEC(N+1-I)
        END DO
      END IF
C
C  Hilbert filters or Quad Part of non-linear phase: duplicate above
C  except use quadrature data.
C
      IF(JTYPE.GT.0) THEN
        WRITE(6,'(X,A11)') ' Quad Part:'
        DO I=1,NFFT
          X(I)=0.0
          Y(I)=TEMPQ(I)
        END DO
        CALL FFT842(1,NFFT,X,Y)
        IF(ODD) THEN            !starting estimate
          J=N
          DO I=1,N
            VEC(J)=X(I)
            J=J-1
          END DO
        ELSE
          DO I=1,N
            VEC(I)=X(I)
          END DO
        END IF
        WRITE(6,'(X,A11)') ' First pass'
        LOOP=3
        CALL VA07A(RESID,LSQ,M,N,VEC,R,SS,A,D,EPS,IPRINT,MAXFN,MODE,
     +             TEMPR,TEMPQ,NFFT)
C
C  stopband pass for quad. part
C
        IF(OPTION.EQ.1) THEN
          WRITE(6,'(X,A19)') ' Stopband only pass'
          LOOP=4
          CALL VA07A(RESID,LSQ,MS,N,VEC,R,SS,A,D,EPS,IPRINT,MAXFN,MODE,
     +               TEMPR,TEMPQ,NFFT)
        END IF
        DO I=1,N
          TEMPQ(I)=VEC(N+1-I)
        END DO
      END IF
C
C  now form impulse response and leave result in working X array with Y=0
C
```

```
      DO I=1,NFFT
        X(I)=0.0
        Y(I)=0.0
      END DO
      IF(ODD) THEN
        X(1)=TEMPR(1)
        IF(JTYPE.EQ.1) X(1)=0.0
        DO I=2,N
          J=NFFT+2-I
          X(I)=TEMPR(I)/2.0-TEMPQ(I)
          X(J)=TEMPR(I)/2.0+TEMPQ(I)
        END DO
      ELSE
        DO I=1,N
          J=NFFT+1-I
          X(I)=TEMPR(I)-TEMPQ(I)
          X(J)=TEMPR(I)+TEMPQ(I)
        END DO
      END IF
      RETURN
      END
C-------------------------------------------------------------------------
C            SUBROUTINE RESID
C  11-SEP-1986
C  User written subroutine required by VA07A
C
C      Called By:  VA07A
C
C      Calls:      nothing
C-------------------------------------------------------------------------
      SUBROUTINE RESID(M,N,VEC,R,IFL,TEMPR,TEMPQ,NFFT)

      REAL      VEC(N), R(M), TEMPR(NFFT), TEMPQ(NFFT)
      REAL      FREQ, SUM, FINCR, WEIGHT(512)
      INTEGER   M, N, IFL
      INTEGER   LOOP, H2IMP, F2TYPE, NF1, NF2, NF3, NF4
      LOGICAL*1 ODD
C
C  /USERA/ shared with modules CUTHBERT and LSQ
C  /USERB/ shared with module CUTHBERT
C
      COMMON/USERA/ DR(512,300)
      COMMON/USERB/ FINCR, LOOP, WEIGHT, ODD, H2IMP, F2TYPE,
     +              NF1, NF2, NF3, NF4
C
C  load DR with dr(k)/dx(i), for k = 0,...,M and i = 0,...,N
C  at the same time as calculate r(i). This is used later in LSQ
C
      FRAC=FLOAT(H2IMP-1)/2.0
      SUBLOOP=2
      IF(LOOP.EQ.1.OR.LOOP.EQ.3) THEN
        NA=1
        NB=NF1
        IF(F2TYPE.EQ.1) SUBLOOP=SUBLOOP+1
      ELSE
        NA=NF2                          !default lowpass stopband
        NB=NFFT/2
```

```
        IF(F2TYPE.EQ.1) THEN
          NA=1                          !lower stopband for bandpass
          NB=NF1
        END IF
      END IF
      L=1
      AMULT=2.0
      IF(ODD) AMULT=1.0
      DO J=1,SUBLOOP
        DO I=NA,NB              !I is frequency increment
          SUM=0.0
          FREQ=FINCR*FLOAT(I-1)
          DO K=0,N-1
            IF(LOOP.LE.2) THEN
              COEFF=AMULT*COS(FREQ*(FLOAT(K)-FRAC))
            ELSE IF(LOOP .GE. 3) THEN
              COEFF=-2.0*SIN(FREQ*(FLOAT(K)-FRAC))
            END IF
            SUM=SUM+VEC(K+1)*COEFF
            DR(L,K+1)=COEFF*WEIGHT(I)
          END DO
          IF(LOOP.LE.2) THEN
            Z=TEMPR(I)
            R(L)=(SUM-Z)*WEIGHT(I)
          ELSE IF(LOOP.GE.3) THEN
            Z=TEMPQ(I)
            R(L)=(SUM-Z)*WEIGHT(I)
          END IF
          L=L+1
        END DO
C
C  set next interval for next subloop if required
C
        NB=NFFT/2
        IF(J.EQ.1) THEN
          NA=NF2                        !default stopband of lowpass
          IF(F2TYPE.EQ.1) THEN
            IF(LOOP.EQ.1.OR.LOOP.EQ.3)THEN
              NB=NF3        !passband for bandpass
            ELSE
              NA=NF4        !upper stopband for bandpass
            END IF
          END IF
        ELSE IF(J.GE.2) THEN            !only has meaning for bandpass
          NA=NF4                        !on LOOP 1,3
        END IF
      END DO
      RETURN
      END
C-----------------------------------------------------------------------
C            SUBROUTINE LSQ
C  11-SEP-1986
C  User written subroutine required by VA07A to set up the coefficients
C  of the least squares normal equations.
C
C      Called By:  VA07A
C
```

```
C      Calls:      FM02AS            (Harwell Library Subroutine)
C-----------------------------------------------------------------------
      SUBROUTINE LSQ(M,N,VEC,R,A,V)

      INTEGER M, N
      REAL*4  VEC(N), R(M), A(N,N), V(N)
      REAL*8  FM02AS
      COMMON/USERA/ DR(512,300)
C
      DO I=1,N
        V(I)=FM02AS(M,DR(1,I),1,R(1),1)
        DO J=1,I
          A(I,J)=FM02AS(M,DR(1,I),1,DR(1,J),1)
        END DO
      END DO
      RETURN
      END
```

```
C-----------------------------------------------------------------------
C  MODULE NAME: SUBROUTINE ALGAZI
C
C  DATE:        1-OCT-1986
C
C  AUTHOR:   R.D.C. STEPHEN
C            DEPT. OF ELECTRICAL AND ELECTRONIC ENGINEERING
C            UNIVERSITY OF CANTERBURY
C            CHRISTCHURCH
C            NEW ZEALAND
C
C  LANGUAGE: VAX FORTRAN, Version 4.3
C            (note: VAX FORTRAN is a superset of FORTRAN-77)
C
C  DESCRIPTION:
C
C       Uses the Iterative technique of V.R. Algazi and Minsoo Suk:
C  "On the Frequency weighted Least-Square Design of Finite Duration
C  Filters"., IEEE Trans Vol CAS-22 No 12 December 1975 pp 943-954.
C  The routine implements equation 35 of the above paper.
C
C -->  As yet, this routine CANNOT handle Hilbert filters. <------
C
C       The routine is setup to use the FFT algorithm FFT842 from
C  the package FAST.FOR in the suite of programs from "Programs for
C  Digital Signal Processing", IEEE Press.
C
C       On entry, the routine expects the ideal frequency specification
C  in the X, Y arrays; the desired impulse response length in HIMP and
C  the weight in WEIGHT.
C       At the start, the routine uses the frequency specification to
C  set up the bandlimiting indicies K1 and K2. Then, if HIMP is even, the
C  frequency spec is offset before computing the model impulse response.
C  Then the model impulse response is computed and held in TEMP. The
C  routine iterates for ITER
C  iterations.
C
C       On exit, the routine leaves the final impulse response in X
C  with Y=0
C
C  USAGE:
C        CALL ALGAZI(X,Y,TEMP,NFFT,HIMP,WEIGHT,ITER,FTYPE,SAFREQ)
C
C  PARAMETERS:
C
C     X,Y    : REAL*4; FFT data arrays of dimension NFFT
C
C     TEMP   : REAL*4; temporary storage; dimension NFFT
C
C     NFFT   : INT*4; dimension of arrays
C
C     HIMP   : INT*4; desired length of impulse response
C
C     WEIGHT: REAL*4; weight parameter required ("A" of equation
C                     35 of paper)
C
C     ITER   : INT*4; number of iterations to perform
```

```
C
C     FTYPE : INT*4; type of filter: 0 = lowpass
C                                    1 = bandpass
C                                    2 = highpass
C
C    SAFREQ : REAL*4; the current sampling frequency. This is only
C                     required by the module EVEN
C
C  SUBROUTINES REFERENCED:
C
C     EVEN
C     FFT842
C
C  NOTES:
C     The data format assumed by the algorithm ignores the linear
C  phase term by expecting the +ve frequency  (time) components
C  in the lower half of the arrays and the -ve frequency (time)
C  components in the upper half.
C  The user will therefore have to manipulate (swap) the data
C  positions before writing out the impulse response.
C
C     The routine writes the iteration number to default system
C  output (UNIT 6). This is the users terminal on a VAX
C
C     The filter frequency specification may be linear or
C  non-linear phase.
C
C  *** no TABS have been used in this code ***
C
C-----------------------------------------------------------------------
      SUBROUTINE ALGAZI(X,Y,TEMP,NFFT,HIMP,WEIGHT,ITER,FTYPE,SAFREQ)
C
      INTEGER*4  NFFT, HIMP, ITER, FTYPE
      INTEGER*4  J, K, K1, K2, M
      REAL*4     X(NFFT), Y(NFFT), TEMP(NFFT), WEIGHT, SAFREQ, Z
      LOGICAL*1  EV
C
      EV=.FALSE.
      IF(MOD(HIMP,2).EQ.0) EV=.TRUE.
      IF(EV) CALL EVEN(X,Y,NFFT,SAFREQ)
C
C  get indicies for bandlimiting template. Assumes passband on input
C  is only region with magnitude > 0 on entry. Look for magnitude so
C  that non-linear phase can be handled.
C
      IF(K3.EQ.0) THEN
      K1=1
      K2=NFFT/2
      Z=0.0
      IF(FTYPE.NE.2) THEN
        DO WHILE(Z.EQ.0.0)
          K2=K2-1
          Z=SQRT(X(K2)**2+Y(K2)**2)
        END DO
      END IF
      K2=K2+1
      Z=0.0
```

```
      IF(FTYPE.NE.0) THEN
        DO WHILE(Z.EQ.0.0)
          K1=K1+1
          Z=SQRT(X(K1)**2+Y(K1)**2)
        END DO
      END IF
      K1=K1-1
C
C  now store model of ideal impulse response. Then zero Y and
C  duration limit to HIMP as starting approximation
C
       CALL FFT842(1,NFFT,X,Y)
       DO I=1,NFFT
         TEMP(I)=X(I)
         Y(I)=0.0
       END DO
       J=HIMP/2+1
       K=NFFT+1-J
       IF(.NOT.EV)J=J+1
       DO I=J,K
         X(I)=0.0
       END DO
       END IF
C
C  begin
C
       Z=(WEIGHT-1)/WEIGHT
       DO M=1,NITER
         WRITE(6,'(A2,I3)') '+ ',M
C
C  get current frequency response of current estimate
C
         CALL FFT842(0,NFFT,X,Y)
C
C  bandlimit current approximation
C
         IF(FTYPE.NE.2) THEN
           K=NFFT-K2+2
           DO I=K2,K
             X(I)=0.0
             Y(I)=0.0
           END DO
         END IF
         IF(FTYPE.NE.0) THEN
           X(1)=0.0
           Y(1)=0.0
           DO I=2,K1
             J=NFFT-I+2
             X(I)=0.0
             Y(I)=0.0
             X(J)=0.0
             Y(J)=0.0
           END DO
         END IF
C
C  now get T-domain of bandlimited approximation, weight it and add
C  weighted sum of ideal over HIMP range
C
```

```
          CALL FFT842(1,NFFT,X,Y)
          K=HIMP/2
          DO I=1,K
            J=NFFT+1-I
            X(I)=X(I)*Z+TEMP(I)/WEIGHT
            X(J)=X(J)*Z+TEMP(J)/WEIGHT
          END DO
          IF(.NOT.EV) X(K+1)=X(K+1)+TEMP(K+1)/WEIGHT
C
C  finally duration limit the result to HIMP and clear Y
C
          J=K+1
          K=NFFT+1-J
          IF(.NOT.EV)J=J+1
          DO I=J,K
            X(I)=0.0
          END DO
          DO I=1,NFFT
            Y(I)=0
          END DO
C
C  loop again
C
        END DO
        RETURN
        END
```

```
C-----------------------------------------------------------------------
C  MODULE NAME: SUBROUTINE EVEN
C
C  DATE:         SEPTEMBER 1986
C
C  AUTHOR:  R.D.C. STEPHEN
C           DEPT. OF ELECTRICAL AND ELECTRONIC ENGINEERING
C           UNIVERSITY OF CANTERBURY
C           CHRISTCHURCH
C           NEW ZEALAND
C
C  LANGUAGE: VAX FORTRAN, Version 4.3
C            (note: VAX FORTRAN is a superset of FORTRAN-77)
C
C  DESCRIPTION:
C      This subroutine offsets the frequency domain data by exp(jwT/2)
C  so that after an inverse FFT, the impulse response is made up of an
C  even number of points - ie the h(0) position does not exist.
C      The routine is designed to cope with linear or non-linear phase
C  filters, so the computations are done on each frequency point i
C  without assuming any symmetry properties at all.
C      This code is designed to go with, and is required by the modules
C  CUTHBERT, DOLPHCHEB, ITERATIVE AND KAISER. Thus, the code is based
C  on the FFT842 code from the suite of programs from "Programs for
C  Digital Signal Processing", IEEE Press, and assumes the same data
C  format in the arrays as the modules that call this one.
C
C      Since the frequency response H(w) is complex, then the new
C  values in the X and Y arrays are:
C
C        H(w)exp(jwT/2) = Hr*cos(wT/2) - Hi * sin(w(T/2)
C                       + j[Hi * cos(wT/2) + Hr * sin(wT/2)]
C
C               Hr = real part of H(w)
C               Hi = imag part of H(w)
C
C       This computation explicitly excludes the DC term (F=0) since
C       cos(0) = 1.0 and sin(0) = 0 so X(w=0), Y(w=0) are unchanged.
C
C  USAGE:
C         CALL EVEN(X,Y,NFFT,SAFREQ)
C
C  PARAMETERS:
C
C      X,Y   :  REAL*4; FFT arrays containing the frequency domain data.
C
C     NFFT   :  INT*4; dimension of the arrays.
C
C    SAFREQ  :  REAL*4; the current sampling frequency
C
C  SUBROUTINES REFERENCED:  none
C-----------------------------------------------------------------------
       SUBROUTINE EVEN(X,Y,NFFT,SAFREQ)
C
       PARAMETER PI=3.1415926
       INTEGER*4 NFFT, K
```

```
      REAL*4    X(NFFT), Y(NFFT), OMEGA, SAFREQ
      REAL*4    TEMP, FRAC
C
      FRAC=2.0*PI*SAFREQ/FLOAT(NFFT)
      K=NFFT+2
      DO I=2,NFFT/2
        OMEGA=FRAC*FLOAT(I-1)/(2.0*SAFREQ)
        TEMP=Y(I)*COS(OMEGA)+X(I)*SIN(OMEGA)
        X(I)=X(I)*COS(OMEGA)-Y(I)*SIN(OMEGA)
        Y(I)=TEMP
        TEMP=Y(K-I)*COS(-OMEGA)+X(K-I)*SIN(-OMEGA)
        X(K-I)=X(K-I)*COS(-OMEGA)-Y(K-I)*SIN(-OMEGA)
        Y(K-I)=TEMP
      END DO
      K=NFFT/2+1
      OMEGA=FRAC*FLOAT(NFFT/2)/(2.0*SAFREQ)
      TEMP=Y(K)*COS(OMEGA)+X(K)*SIN(OMEGA)
      X(K)=X(K)*COS(OMEGA)-Y(K)*SIN(OMEGA)
      Y(K)=TEMP
      RETURN
      END
```

APPENDIX 7

The VAX FORTRAN subroutines UNWRAP and WRAP used to implement the phase unwrapping and phase wrapping procedures shown in Fig. 6.11.

```
C-----------------------------------------------------------------------
C  MODULE NAME: SUBROUTINE UNWRAP
C
C  DATE:         NOVEMBER 1986
C
C  AUTHOR:       R.D.C. STEPHEN
C                DEPT. OF ELECTRICAL AND ELECTRONIC ENGINEERING
C                UNIVERSITY OF CANTERBURY
C                CHRISTCHURCH
C                NEW ZEALAND
C
C  LANGUAGE:     VAX FORTRAN, Version 4.4
C
C  DESCRIPTION:
C
C     Performs the phase unwrapping procedure of Fig. 6.11(a). The
C  routine is set up for division by 2, so the JSIZE=ISIZE/2.
C     The code uses the basic methodology for implementing the
C  convolution relation described in Fig. 5.3 and which was designed
C  to eliminate the (usual) nfilter/2 delay between the input and
C  output. Thus the first and last NFILTER/2 samples of OUTPUT
C  must be discarded.
C
C  USAGE:  CALL UNWRAP(NFILTER,FILTCOEFF,ISIZE,JSIZE,INPUT,OUTPUT)
C
C  PARAMETERS:
C
C     NFILTER:   INT*4; length of filter impulse response
C
C     FILTCOEFF: REAL*4; array holding filter coefficients
C
C     ISIZE:     INT*4; no of samples in input data
C
C     JSIZE:     INT*4; no of samples in output array
C
C     INPUT:     REAL*4; array of dimension ISIZE with input data
C
C                        to the filter.
C
C     OUTPUT:    REAL*4; array of dimension JSIZE with eventual output
C                        data
C
C
C  SUBROUTINES REFERENCED:  none
C-----------------------------------------------------------------------
      SUBROUTINE UNWRAP(NFILTER,FILTCOEFF,ISIZE,JSIZE,INPUT,OUTPUT)
C
      INTEGER NFILTER, ISIZE, JSIZE
      INTEGER I, IJ, J, K, L, M, N
      REAL    FILTCOEFF(NFILTER), INPUT(ISIZE), OUTPUT(JSIZE),
     +        SAMPLE, SUM
C
      L=NFILTER/2       !start with the half the window into the data.
      IF(MOD(NFILTER,2) .GT. 0) L=L+1       !if filter has odd length
      I=1
      J=1
      N=NFILTER-1
      M=1
```

```
      DO WHILE(I.LE.JSIZE)       !continue for whole output array length
        SUM=0.0
C convolution loop: do it every 2nd input sample
        IF(M.EQ.1) THEN
          DO K=0,N
            IJ=L-K
            SAMPLE=0.0
            IF(IJ.GE.1 .AND. IJ.LE.ISIZE) SAMPLE=INPUT(IJ)
            END IF
            SUM=SUM+SAMPLE*FILTCOEFF(J)
            J=J+1
            IF(J.GT.NFILTER) J=1       !J is modulo NFILTER
          END DO
          OUTPUT(I)=SUM
          I=I+1                      !next output sample pointer
C
C J leaves the convolution loop with the value it started, hence
C decrement causes a shift (of centre point of the filter) by one
C sample to the right.
C
          J=J-1                      !shift filter index pointer
          IF(J.LE.0) J=NFILTER       !J is modulo NFILTER
        ELSE
C second pass through, just reset M
          M=0
        END IF
        M=M+1           !increment convolution loop index pointer
        L=L+1           !increment input sample pointer each time
      END DO
      RETURN
      END
C-------------------------------------------------------------------------
C  MODULE NAME: SUBROUTINE WRAP
C  DATE:        NOVEMBER 1986
C
C  AUTHOR:      R.D.C. STEPHEN
C               DEPT. OF ELECTRICAL AND ELECTRONIC ENGINEERING
C               UNIVERSITY OF CANTERBURY
C               CHRISTCHURCH
C               NEW ZEALAND
C
C  LANGUAGE:    VAX FORTRAN, Version 4.4
C
C  DESCRIPTION:
C
C     Performs the phase wrapping procedure of Fig. 6.11(b). The
C  routine is set up for multiplication by 2, so JSIZE=2*ISIZE
C     The code uses the basic methodology for implementing the
C  convolution relation described in Fig. 5.3 and which was designed
C  to eliminate the (usual) nfilter/2 delay between the input and
C  output. Thus the first and last NFILTER/2 samples of OUTPUT
C  must be discarded.
C
C  USAGE:  CALL WRAP(NFILTER,FILTCOEFF,ISIZE,JSIZE,INPUT,OUTPUT)
C
C  PARAMETERS:
C
C     NFILTER:   INT*4; length of filter impulse response
```

```
C
C     FILTCOEFF: REAL*4; array holding filter coefficients
C
C     ISIZE:     INT*4; no of samples in input data
C
C     JSIZE:     INT*4; no of samples in output array
C
C     INPUT:     REAL*4; array of dimension ISIZE with input data
C
C                       to the filter.
C
C     OUTPUT:    REAL*4; array of dimension JSIZE with eventual output
C                       data
C
C
C  SUBROUTINES REFERENCED:  none
C-----------------------------------------------------------------------
      SUBROUTINE WRAP(NFILTER,FILTCOEFF,ISIZE,JSIZE,INPUT,OUTPUT)
C
      INTEGER NFILTER, ISIZE, JSIZE
      INTEGER I, IJ, J, K, L, M, N
      REAL    FILTCOEFF(NFILTER), INPUT(ISIZE), OUTPUT(JSIZE),
     +        SAMPLE, SUM, PI2
C
      L=NFILTER/2       !start with the half the window into the data.
      IF(MOD(NFILTER,2).GT.0) L=L+1       !if filter has odd length
      I=1
      J=1
      N=NFILTER-1
      DO WHILE(I.LE.JSIZE)  !continue for length of output array
C
C  do twice for every one input sample
C
        DO M=1,2
          SUM=0.0
C
C  convolution loop
C
          DO K=0,N
            IJ=L-K
            SAMPLE=0.0
            IF(IJ.GE.1 .AND. IJ.LE.ISIZE) SAMPLE=INPUT(IJ)
            SUM=SUM+SAMPLE*FILTCOEFF(J)
            J=J+1
            IF(J.GT.NFILTER) J=1
          END DO
          OUTPUT(I)=SUM
          I=I+1
          IF(M.EQ.1) THEN
            J=J+1                     !shift filter index pointer
            IF(J.GT.NFILTER) J=1      !J is modulo NFILTER
          END IF
        END DO
        L=L+1        !increment input pointer after every 2 loops
      END DO
      RETURN
      END
```

ADDENDA

1. Page 19, line 1: [Woodward...]

An examiner recommended Woodward as the reference for the discussion on the uncertainty relation.

However, while it is acknowledged that Woodward predates Temes, Woodward's quotation in Eqn 8, p119 is based on radar in relation to ambiguity (uncertainty) between range and velocity.

In the context of this thesis, I think Woodward's presentation may lead to confusion. The reader may also consider Papoulis [1962a], p63 which is presented in terms much closer in context to the present discussion.

2. Page 100, line 25: "... it is physically impossible..."

This statement is wrong. It is quite simple to construct a case where both $s(t)$ and $\hat{s}(t)$ are zero simultaneously. eg

$$s(t) = \sin(x) - \sin(2x) \qquad \text{(Add.1)}$$

The Hilbert Transform of Equation (Add.1) is

$$\hat{s}(t) = -\cos(x) + \cos(2x) \qquad \text{(Add.2)}$$

Both equations (Add.1) and (Add.2) are zero for $x=2n\pi$, $n=0,1,2,...$

The discussion in the thesis is badly phrased. THe example above is considered a special case and not representative of real speech.

Anticipating the introductory comments in Chapter 5, it is known from the work of Gold [1964], Golden [1968] and Kang et al [1983], related to vocoders, that introducing a speech-like phase (some randomness in the starting phase values) gives perceptually better synthesised speech.

The corollary to this is that the traditional vocoder model of synthesised voiced speech

$$s_{syn}(t) = \sum_{k=1}^{n} A_k \sin(k\omega t) \qquad \text{(Add.3)}$$

$$A_k = 1, \quad k=1,...,n$$

is poor and that voiced speech is more accurately represented as

$$s_{syn}(t) = \sum_{k=1}^{n} A_k \sin(k\omega t + \theta_k) \qquad \text{(Add.4)}$$

where the θ_k are the phase values of the harmonics.

Thus, a more accurate comment should be that for large n, and recognising that the vocal tract is in constant motion, the probability that both equation (Add.4) and its Hilbert Transform are exactly zero simultaneously is very small, but finite. Thus, for a real speech signal, the analytic zeros $\{x_k\}$ are almost entirely complex. Occaisionally, a single analytic zero will approach the real axis reflecting that both s(t) and $\hat{s}(t)$ are nearly zero simultaneously. Fig. 5.6(c) is a good example.

3. Page 121, line 33: "This value of peak factor...."

This statement is wrong. In fact a square-wave of amplitude ±1 has a peak factor of 2. However, a square wave is a special case of a sum of odd order harmonics with a special power spectrum (ω_o, $1/3\omega_o$, $1/5\omega_o$ etc.) and therefore not representative of a speech signal.

In this context, a paper by Schroeder that was missed and has a direct bearing on this issue is

SCHROEDER, M.R., (1970) Synthesis of Low Peak-Factor Signals and Binary Sequences with Low Autocorrelation. IEEE Trans. Inf. Theory Vol IT-16 January 1970 pp 85-89.

4. Page 124, line 22: "7. The irreducible..."

In view of the erroneous statement in 3. above, this paragraph should be deleted.

5. Page 129, line 25: "The important points of..."

Insert before this line:

Note that by definition from equation (5.7), PF_{seg} is a dimensionless quantity. Thus the abscissa of Fig. 5.9(c),(d) and Figs 5.10 - 5.12 are dimensionless.